EPSON

日本語DiskBASIC

user's manual

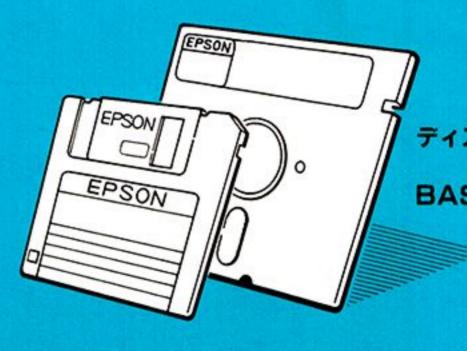
ユーザーズマニュアル●

解説編

BASICの起動と終了
BASICの文法
キーボードとスクリーンエディタ
日本語入力
入出力装置とファイル
テキスト画面とグラフィック画面
プリンタ
データ通信
機械語モニタ
機械語プログラム
データの内部構造
マウス
ディスクのファイル管理

BASICユーティリティ

付録

PC286ML9
Y16199113601

ご注意

(1) 本書の内容の一部、または全部を無断で転載することは、固くお断りします。

(2) 本書の内容について、将来予告なしに変更することがあります。

(3) 本書の内容については万全を期して作成致しましたが、万一誤り・お気付きの点がありましたら、ご連絡くださいますようお願いいたします。

(4) 運用した結果の影響については、(3)項にかかわらず責任を負いかねますのでご了承ください。

本製品を日本国外へ持ち出す場合のご注意

本製品は「外国為替及び外国貿易管理法」に定める戦略物資(または役務)に該当します。したがって、本製品を輸出する場合には同法に基づく日本国政府の輸出許可が必要です

EPSON

日本語DiskBASIC

user's manual

ユーザーズマニュアル●

はじめに

日本語 Disk BASIC

日本語 Disk BASIC は、基本的な BASIC をもとに強力な日本語処理機能、グラフィック機能、スクリーンエディタ機能、倍精度計算機能などを拡張した BASIC です。日本語 Disk BASIC を理解することによって、自分で自由にプログラムを作成して、さまざまな処理を行わせることができます。

日本語 Disk BASIC のマニュアルは次の2冊から構成されています。

日本語 Disk BASIC ユーザーズマニュアル

日本語 Disk BASIC リファレンスマニュアル

日本語 Disk BASIC ユーザーズマニュアル

日本語 Disk BASIC ユーザーズマニュアルは日本語 Disk BASIC を機能ごとに解説したもので、次の3部構成になっています。

解説編

日本語 Disk BASIC でプログラムを作成するのに必要な事柄を機能別に解説しています。具体的には日本語 Disk BASICの起動と終了の方法、プログラム作成のための基礎知識、キーボードやグラフィック機能などを解りやすく説明しています。

BASIC ユーティリティ編

日本語 Disk BASIC ユーティリティディスクに入っているユーティリティソフト「BMENU」、「BTERM」、「SYSSET」の使用方法について説明しています。BMENU によってディスクのフォーマットやメモリスイッチの変更などを行うことができます。

付録

日本語 Disk BASIC の補足的な技術資料をまとめて説明しています。PC-286U/US シリーズのサウンド機能についても解説しています。

日本語 Disk BASIC リファレンスマニュアル

日本語 Disk BASIC リファレンスマニュアルは、日本語 Disk BASIC を構成する命令を具体的に解説したものです。BASIC の命令にはコマンド・ステートメント・関数の区別があり、これらを、引きやすさ、探しやすさを重視して、アルファベット順に説明しています。また、機能面からの検索の際に役立つように、機能別の索引もつけました。それぞれのケースに応じて活用してください。

これらのマニュアルは EPSON PC シリーズすべての共通マニュアルになっており、機種によっては使用できない機能も一緒に解説していますのでご注意ください。このような機能には次のものがあります。

サウンド機能	PC-286U/US シリーズで使用可能です。
数値演算プロセッサ	PC-286L/LE シリーズ以外の機種は、オプションで数値演算プロセッサを取り付けることができます。

目　　次

解説編

第1章

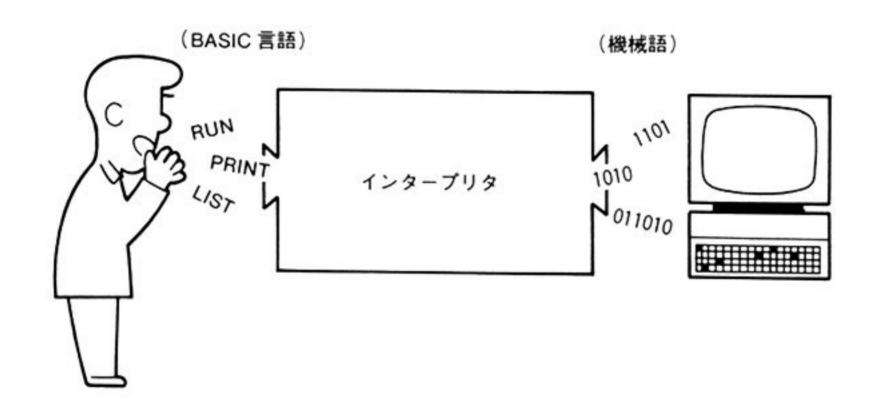
BASICの起動と終了

日本語 Disk BASIC(以後 BASIC と記述)の起動方法と終了方法、さらに BASIC を使用する上で必要な基本的なことがらについて説明します。

1.1　BASICとは

コンピュータは、機械語と呼ばれる言葉しか理解することができません。その機械語は1と0との集まりで表現した言葉で、私たちがふだん使っている言葉とはまったく性質の異なるものです。そのため機械語を直接使ってコンピュータに指示をすることはとても大変です。

「人間が普段使っている言葉に、より近い表現でコンピュータに指示できないものだろうか」そんな願いから BASIC 言語が開発されました。

本書で説明する日本語 Disk BASIC は、正確にはインタープリタ方式の BASIC 言語といいます。インタープリタとは「通訳」のことで、私たちが機械語を知らなくとも、BASIC で表現された指示の内容を解釈(インタープリット)し、コンピュータに伝えます。

BASIC とは Beginner's All-purpose Symbolic Instruction Code の略で、「初心者向きの多目的に使える記号で表した命令語」と訳すことができます。このことからもわかるように、アメリカで生れたコンピュータ言語で、使用する言葉もすべて英語です。

しかし RUN、LIST、SAVE、STOP といったように簡単で基本的な単語がほとんどです。英語が不得意という人でも十分理解できます。

1.2 システムディスク

日本語 Disk BASIC には、2枚のシステムディスクがついています。

日本語 Disk BASIC システムディスク
日本語 Disk BASIC ユーティリティディスク

日本語 Disk BASIC システムディスク
どちらのディスクでも BASIC を起動することができます。
日本語 Disk BASIC システムディスクは、BASIC を起動するためのプログラム(システムプログラム)のほかに、かな漢字変換方式を用いて日本語入力を行うための、日本語入力システムが入っています。
このディスクには次のようなファイルが入っています。

```
egbdic man      egbdic usr      egzipm dic      egzipu dic
```

注意 システムプログラムや、日本語入力システムはファイルとしてディスクに書き込まれていないため、FILESコマンドなどで確かめることはできません。
BMENU の「フォーマット」の中で「システムディスク作成」を実行することによりディスクの特別な場所に書き込まれます。

日本語 Disk BASIC ユーティリティディスク

ユーティリティディスクには、システムプログラムのほかに、ディスクをフォーマットしたり、メモリスイッチの内容を変更したりするためのユーティリティプログラムが入っています。

```
BMENU  .      bmenu  .      BMENU  .AUT   BMENU  .BKP   bmenu  *bkp
BMENU  .CPY   bmenu  *cpy   BMENU  .DCM   bmenu  *dcm   bmenu  *ent
BMENU  .FMT   bmenu  *fmt   BMENU  .HCP   BMENU  .HDM   bmenu  *hdb
bmenu  *hdm   BMENU  .MKF   bmenu  *mkf   BMENU  .SET   BMENU  .SSW
BMENU  .ZPD   BMENU  .ZPM   BTERM  .      BTERM  .EDT   BTERM  .SET
BTERM  .TRM   mouse  *cod   SYSSET.
```

注意 　フロッピーディスクは、カセットテープと同じ磁気記録媒体のひとつです。磁石など磁気を発生しているものを近付けると、重要なデータが消えてしまうことがあります。また、フロッピーディスクは消耗品ですから、何回もデータを読み書きしていると、正常に読み書きができなくなってくることもあります。このためシステムディスクなど重要なデータの入っているディスクは、同じ内容を記録した予備のディスクを作るようにします。
　このように、もとのディスクと同じ内容のディスクを作ることをバックアップをとるといいます。システムディスクは必ずバックアップをとり、ふだんは、このバックアップしたディスクを使用するようにします。そして、もとのディスクは大切に保管しておくようにしてください。バックアップの方法は「1.8　システムディスクのバックアップ」を参照してください。

1.3　BASIC の起動

BASIC のシステムプログラムをコンピュータのメモリに読み込ませて、BASIC を使える状態にすることを BASIC を起動する、または立ち上げるといいます。

BASIC を起動するには、二通りの方法があります。一つはシステムディスクをフロッピーディスクドライブにセットする方法、もう一つはシステムディスクの内容をハードディスクに記録して、ハードディスクから起動する方法です。ハードディスクから起動する方法は「1.10　BASIC とハードディスク」を参照してください。

① 電源スイッチをオンにして、システムディスクをドライブ1にセットします。ドライブ2にセットしても構いません。電源スイッチがすでにオンになっている場合は、システムディスクをセットして、リセットボタンを押します。

② システムディスクをセットしたドライブのアクセスランプが点滅して、システムプログラムが読み込まれます。

③ さらに時間が経過すると、アクセスランプが消灯して、画面に次のようなメッセージが表示されます。

```
Disk version
How many files?(0-15)              ............③-1
EPSON 日本語 BASIC ver X. X        ............③-2
Copyright (c)SEIKO EPSON CORP. 1987, Revised Edition 1988
 XXXXXX Bytes free                 ............③-3
Ok
```

③-1 プログラム実行中に、同時にオープンする必要のあるファイルの数を指定します。［↵］キーだけを押すと 2 *を指定したのと同じ意味になります。もし、必要なファイルの数が 2 でなければ必要な数を入力して［↵］キーを押します。入力できる最大の数は15です。0 を入力するとプログラム中でファイルを扱うことができなくなります。

* 実際は接続されているフロッピディスクドライブとハードディスクドライブの合計の台数の数になります。

③-2 X. X は日本語 Disk BASIC のバージョンを示します。

③-3 XXXXXX は BASIC のユーザーエリアの大きさを示します。次の要因で、この大きさが変わってきます。

- 日本語入力の拡張　（「第4章　日本語入力」参照）
- 同時にオープンするファイルの数（「第5章　入出力装置とファイル」参照）
- カラー表示の拡張　（「第6章　テキスト画面とグラフィック画面」参照）
- ハードコピーの拡張　（「第7章　プリンタ）
- 機械語モニタの拡張　（「第9章　機械語モニタ」参照）

④ これで BASIC を使用する準備ができました。

1.4　BASICの終了とリセット

BASICを終了する場合は、次の点を確認して電源スイッチをオフにしてください。

- メモリ上にある必要なプログラムをフロッピーディスクやハードディスクにセーブします。
- ハードディスクを接続している場合は、必ずCLOSE文を実行して磁気ヘッドの退避処置(リトラクト、シップディスク)を行ってください。

以上の確認をすませフロッピーディスクドライブからフロッピーディスクを抜き出し、コンピュータの電源をオフにします。

このほか、リセットボタンを押すと(リセット)、電源をオンにした直後と同じ状態にすることができます。リセットは電源スイッチをオフにしたのと同じように、メモリ上のデータはすべてクリアされますので使用していたプログラムやデータは消えてしまいます。リセットは次のような場合に使用すると便利です。

(1)　メモリスイッチの内容を変更した場合

(2)　別のシステムを新たに起動したい場合

例えば、BASICを終了して、引き続き日本語ワードプロセッサのソフトを使用しようとする場合など

(3)　何らかの原因でプログラムが暴走して、キー入力ができなくなった場合

1.5　ウォームスタート

[STOP]キーを押しながら、リセットボタンを押すとウォームスタートとよばれる初期化を行うことができます。

ウォームスタートを行うと、画面をクリアし左上にOKと表示します。しかも、リセットと違いメモリ中のプログラムやデータを消去しません。何らかの理由でプログラムが暴走したりして、キー入力ができなくなった場合などに、ウォームスタートを行うともとの状態に戻ることがあります。ただし、この場合はプログラムの暴走によってシステムのデータが破壊されているかもしれませんので、必要なプログラムなどはディスクにセーブして、システムをリセットするようにしてください。

1.6 ディスクドライブとBASIC

BASICでは、フロッピーディスクドライブやハードディスクドライブに数字の名前を付けて、一台一台を区別しています。この数字はドライブ番号またはデバイス名などと呼ばれます。プログラム中でディスクドライブを指定する場合や、ユーティリティソフトBMENUなどでディスクドライブの番号を入力する場合は、このドライブ番号を使います。

BASICはディスクドライブを最大10台まで使用できますので、1から10までのドライブ番号が順番に割り当てられていきます。デバイス名を割り当てる順番はディップスイッチSW1-4とメモリスイッチSW5-1の設定により変更することができます。ここではフロッピーディスクドライブが4台、ハードディスクドライブが2台までの例を説明します。

ディップスイッチSW1-4	OFF	OFF	OFF	OFF	OFF	OFF	ON	OFF	OFF
メモリスイッチSW5-1	0	0	0	1	1	0	0	0	1
設定 / ディスクドライブ	ドライブ番号								
本体フロッピーディスクドライブ1	1	1	1	2	2	1	3	1	3
本体フロッピーディスクドライブ2	2	2	×	3	×	2	4	2	4
増設フロッピーディスクドライブ1	×	×	×	×	×	3	1	3	5
増設フロッピーディスクドライブ2	×	×	×	×	×	4	2	4	6
ハードディスクドライブ1	×	3	2	1	1	×	×	5	1
ハードディスクドライブ2	×	×	×	×	×	×	×	6	2

数字はドライブ番号、×は接続していないことを示します。

ディップスイッチSW1-4の初期設定はOFF、メモリスイッチSW5-1のシステム設定値は0です。したがって、この設定の場合には、本体のフロッピーディスクにドライブ番号1，2が割り当てられます。マニュアルでは、特に指定のない限り、この状態になっているものとして説明しています。

例

RUN "BMENU" ⏎

標準設定の場合であれば、フロッピーディスクドライブ1に入っているディスクのBMENUというプログラムを実行します。

OPEN "2：ABCDEF. DAT " AS ＃1

標準設定の場合であれば、フロッピーディスクドライブ2に入っているディスクの"ABCDEF. DAT"というファイルをオープンします。

デバイス名はプログラム中でどのディスクのどのファイルのデータを読み書きするのかを指定するものです。BASICはディスクドライブのほか、通信回線やプリンタなどの周辺機器もデバイス名で区別します。詳細については「4.1 ファイル」を参照してください。デバイス名を省略した場合は1を指定したことになります。

例 通信回線のデバイス名　　COM1

1.7 フロッピーディスクのフォーマット

購入したばかりのフロッピーディスクや、別のシステムで使用していたフロッピーディスクを使うときには、最初にフォーマットと呼ばれる初期化を行います。フォーマットを行うと、データの読み書きができるようにディスクに特別な情報が書き込まれます。フォーマットしていないディスクに対して、データの読み書きはできません。

BASICで使用するフロッピーディスクはユーティリティソフトBMENUの「フォーマット」でフォーマット処理をしたものを使用します。これ以外のディスクは使用できません。

「フォーマット」には「データディスク作成」、「システムディスク作成」の2つがあります。「システムディスク作成」でフォーマットしたディスクにはBASICのシステムプログラムが書き込まれます。このディスクはBASICの起動用のディスク(システムディスク)として使用できます。「データディスク作成」でフォーマットしたディスクにはシステムプログラムは書き込まれません。システムプログラムは約160KBの容量がありますので、その分、データを記録する容量が少なくなります。

「データディスク作成」でフォーマットしたディスクにも、ファイルをセーブする前であれば「ディスクコードコピー」でシステムプログラムを書き込むことができます。そのディスクにファイルがあるとエラーとなり実行できないことがあります。

1.8 システムディスクのバックアップ

システムディスクはBASICを起動するための大切なディスクです。このような大切なディスクは、バックアップを取り、保管しておきます。そして、ふだんはコピーしたディスクを使用するようにしてください。

バックアップは、添付されている日本語Disk BASICユーティリティディスクに入っているユーティリティソフトBMENUを使って行います。

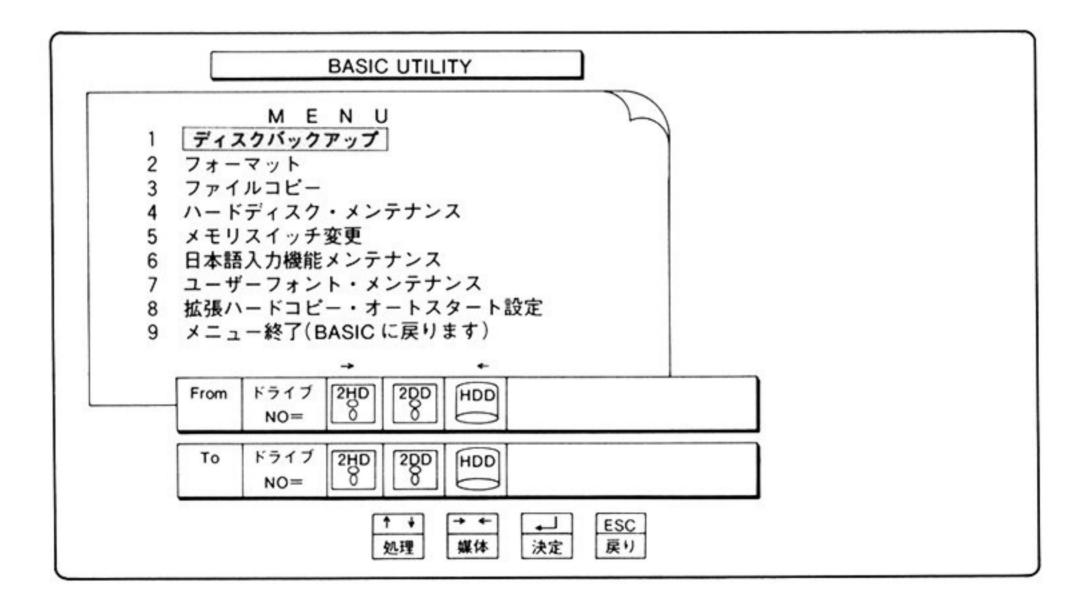
用意するもの

(1) 日本語Disk BASICシステムディスク
日本語Disk BASICユーティリティディスク

注意 誤って、フロッピーディスクのデータを破壊しないように、書き込み禁止の設定をしてください。

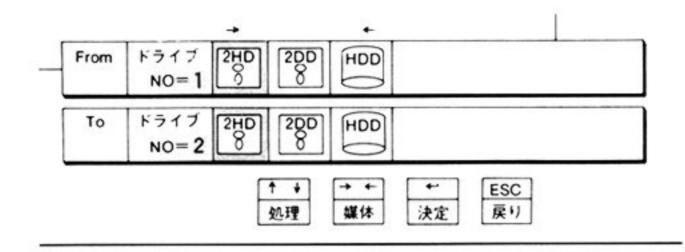
- 5.25インチフロッピーディスク　　ライトプロテクトシールを貼る
- 3.5インチフロッピーディスク　　ライトプロテクトタブを動かす

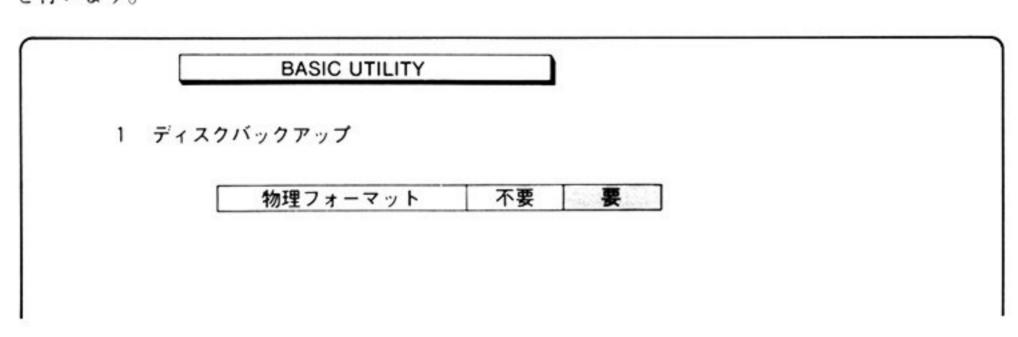
(2) 新しいフロッピーディスク2枚
2HDと呼ばれる1MBタイプのもの

① メモリスイッチをシステム設定値に戻すために、ディップスイッチSW2-5をOFFにして、電源スイッチをオンにします。
② ユーティリティディスクを、ドライブ1にセットして、BASICを起動します。
③ RUN"BMENU" [↵] と入力して、BMENUを実行します。
次のようなメインメニューを表示します。

④ ディスクのバックアップを選びます。[1]キーを押すと、「1　ディスクバックアップ」が反転表示に変わります。

[↵]キーを押すと、「From」と「ドライブNO＝」の入力欄が反転表示に変わります。

⑤ ドライブ番号として1を入力します。フロッピーディスクの種類は2HDですから、カーソル移動キー[→]でディスクの反転表示を2HDと書かれたディスクに移動して、[↵]キーを押します。

⑥ 「To」と「ドライブNO＝」の入力欄が反転表示に変わます。[2]を入力して[↵]キーを押します。これでドライブ1からドライブ2にバックアップコピーをする準備ができました。

⑦ 画面が次のように変わります。

ドライブ2にセットするディスクは新しいものですから、最初にフォーマットする必要があります。カーソル移動キー[→]で物理フォーマットの「要」の位置を反転表示にし、[↵]キーを押します。この設定により、フォーマットを行ってから、バックアップコピーを行います。

⑧ ドライブ2に新しいディスクをセットして、[f・5]キーを押します。処理トラック数154を表示し、フォーマットしているトラックの番号を「フォーマットトラック」の欄に表示します。続いてバックアップコピーを開始し、「コピートラック」の欄にコピーしているトラックの番号を表示します。

⑨　正常に終了すると、ブザーが鳴り「処理終了」と表示します。これで、日本語 Disk BASIC ユーティリティディスクのコピーができました。ドライブ1とドライブ2からディスクを取り出してください。

⑩　次に日本語 Disk BASIC システムディスクのコピーを行います。 ESC キーを押すと、⑦の画面に戻ります。

⑪　ドライブ1に日本語 Disk BASIC システムディスク、ドライブ2に新しいディスクをセットしてください。

⑫　ドライブ2のディスクは新しいディスクですからフォーマットが必要です。「要」の位置を反転表示にして ↵ キーを押します。

⑬　 f・5 キーを押すと処理を開始します。

⑭　正常に終了すると、ブザーが鳴り「処理終了」と表示します。これで、日本語 Disk BASIC システムディスクのコピーができました。ドライブ1とドライブ2からディスクを取り出してください。

⑮　ドライブ1に日本語Disk BASIC ユーティリティディスクをセットして、 ESC キーを2回押してください。③の画面に戻りますので 9 キーを押して、反転表示を「9　メニュー終了」に移動して、 ↵ キーを押します。これで BMENU の実行は終了です。

⑩　これからは、ここでコピーしたディスクをシステムディスクとして使用してください。添付されていたシステムディスクは大切に保管してください。

注意　フロッピーディスクドライブが1台しかないコンピュータでは、次のようにしてバックアップを行ってください。

①　あらかじめ新しいディスクをフォーマットします。これは BMENU の中で「フォーマット」の「データディスク作成」を実行します。

②　BMENU の「ディスクバックアップ」で「From」と「To」のドライブ番号に同じ番号(普通は1)を入力します。

③　「転送元ディスク」(添付のシステムディスク)と「転送先ディスク」(新しいディスク)の指示に従ってディスクを交換しながらコピーを行います。

1.9 プログラムの作成と実行

1.9.1 ダイレクトモードとプログラムモード

BASICを立ち上げるとBASICを使える準備が整ったことになります。いよいよBASICでプログラムを組んでいきますが、動作モードには次のようなものがあります。

ダイレクトモード(直接モード)

行番号をつけずにBASICの文法に従ってコマンドを入力し、[↵]キーを押すと、直ちに実行されます。

例

```
PRINT 25*14
 350
OK
```

[P][R][I][N][T][ ][2][5][*][1][4][↵]

ほとんどのコマンドは、ダイレクトモードで使用できます。

プログラムモード(間接モード)

行番号を付けてコマンドを入力した場合は、その内容を本体内のメモリに、行番号とともに記憶します。そのプログラムはRUNの入力により実行を開始します。

例

```
100  PRINT 25*14
RUN
 350
OK
```

[1][0][0][ ][P][R][I][N][T][ ][2][5][*][1][4][↵]
[R][U][N][↵]

1.9.2 エラーメッセージの表示

プログラム中に間違い(エラー)があると、指示の内容が正しくコンピュータに伝わらず、正常な実行が行えません。BASICがプログラムを実行していく際に、文法上の間違いなどがあった場合には次のようなエラーメッセージを表示して実行を終了します。

- Syntax error/(文法の間違い)

コマンドや関数などが正しいつづりによって入力していないなどの文法上の間違いがあった場合に表示します。Syntaxとは構文、規則などの意味です。

- Illegal function call/(違法関数呼び出し)

負の平方根を求めようとしたなど入力された値が正しい範囲にないときに表示します。

このほか、BASICはプログラムを実行しながら正常に処理できない命令を発見すると、実行を強制的に終了し、エラーメッセージを表示します。プログラム実行中にエラーが発生すると、プログラムの実行は中断します。このとき、エラーの種類とエラーの起きた行番号を表示します。HELP キーを押すと、その行の内容を表示しカーソルはエラーのある項目か、その次の項目で点滅します。エラーによっては、最初からエラーのある行の内容が表示されます。間違いを訂正して ↵ キーを押します。エラーの原因が、表示された行でなく他の行にある場合は、EDITコマンドで、その行を表示させて、エラーの訂正を行います。

EDIT　エラーのあった行番号 ↵

EDITコマンドを入力すると、指定した行を画面最上行に表示しエディットモードになります。ROLL UP キーと ROLL DOWN キーで画面をスクロールできます。

注意 | 1つの行内で修正が終わったら、次の行の修正や入力にはいる前に、必ず ↵ キーを押してください。カーソルは行の、どの位置にあってもかまいません。

エラーメッセージが表示された場合はエラーの発生した原因を取り除いて(プログラムの修正)、再度実行してください。エラーメッセージとエラーの原因については「日本語Disk BASICリファレンスマニュアル」を参照してください。

1.9.3 プログラムのセーブ

プログラムモードで作成したコマンドの集り(プログラム)は、SAVEコマンドでディスクに保存(セーブ)し、後で再び実行することができます。逆にセーブしたプログラムをメモリ中に持ってくる(ロード)コマンドはLOADコマンドとRUNコマンドです。

SAVEコマンドはプログラムをディスクに保存するコマンドです。

SAVE　ファイル名 [, | A |]
| P |

ファイル名に続けてAを指定すると、プログラムをアスキー形式で保存します。これに対してファイル名の後ろに何も付けない場合にはバイナリー形式で保存します。アスキー形式はプログラムを画面に表示する文字のままの形式で保存します。バイナリー形式は、プログラムをそのまま記録するのではなく、中間コードに変換してプログラムを圧縮した形でディスクにセーブします。ですからバイナリー形式で保存するとディスク上の使用領域が少なくてすみます。

ファイル名の後ろにPを付けてプログラムをセーブすると、プロテクトセーブと呼ばれる保存形式になります。プロテクトセーブしたプログラムに対しては、そのままプログラムをメモリにロードしても内容をLISTコマンドで見たり、EDITコマンドで修正することができなくなります。またプロテクトを解除することもできませんので、特別な場合以外使用しないでください。

例 | SAVE"TEST"　メモリ上のプログラムをTESTというファイル名でセーブします。

1.9.4 プログラムのロードと実行

セーブしたプログラムをメモリにロードして、実行するには次のコマンドを実行します。

LOAD ファイル名 [, R]

ファイル名で指定したプログラムをロードします。ロードした後に直ちにプログラムを実行する場合はRオプションを付けます。この場合、そのときオープンしていたファイルはクローズされません。共通のデータファイルを使用している別のプログラムを、続けて実行する場合などに指定します。

例 | LOAD "TEST"　　　　プログラム TEST をロードします。

RUN ファイル名 [, R]

ファイル名を省略するとメモリ上のプログラムを実行しますが、ファイル名を指定することによってプログラムをロードして実行することができます。Rオプションを付けると、そのときオープンしているファイルはクローズされません。これは **LOAD ファイル名,R** を実行したときと同じです。Rオプションを省略するとファイルをすべてクローズします。

例 | RUN"TEST"　　　　プログラム TEST をロードして実行します。

1.10 BASICとハードディスク

ハードディスクを使用する場合の手順を説明します。フロッピーディスクと同じようにフォーマットを行う必要があります。

1.10.1 ハードディスクのフォーマット

購入したばかりのフロッピーディスクや、MS-DOSで使用していたフロッピーディスクを使うときには、最初にフォーマットと呼ばれる初期化を行います。同じようにハードディスクの場合も、初めて使用する場合はフォーマットを行います。

ハードディスクのフォーマットは、フロッピーディスクの場合と異なり、物理フォーマットと論理フォーマットの2段階に分けて行います。また、20MB以下のハードディスクドライブをフォーマットする場合と、40MBのハードディスクドライブをフォーマットする場合の方法が異なりますので注意してください。

1.10.2 20MBハードディスクの使用方法(標準フォーマット)

20MB以下の記憶容量を持つハードディスクに対しては標準フォーマットを行います。

① BMENUの「ハードディスクメンテナンス」の「全領域初期化」で標準フォーマットで初期化を行います。このフォーマットを行うとハードディスク上のすべてのデータが消去されますので注意してください。MS-DOSなどほかのシステムでフォーマットしてある場合は実行する必要はありません。ハードディスク全体をフォーマットすることを物理フォーマットなどともいいます。

② BMENUの「フォーマット」でBASICの領域を確保し、その領域をBASICで使用できるようにフォーマットします。BASIC以外の領域には影響を与えません。フロッピーディスクは1枚ごとに、これはBASIC用、これはアプリケーションソフト用というように区別して使いますが、ハードディスクではこのようなことはできません。その代わり内部的に半分はBASIC、半分はアプリケーションというように分割して使用することができます。このようにシステムごとに使用する領域を確保することを領域確保、あるいは論理フォーマットなどといいます。

1MB単位でハードディスクの大きさまで確保できますので、必要な大きさを確保します。

また、BASICの領域は一番最初に確保するようにしてください。そうしないとBASICの領域が認識されません。

③ BMENUを終了し、リセットを行い再度BASICを立ち上げ直します。この時点からハードディスクが正常に認識され使用することができます。

ハードディスクを接続していると、起動時のメッセージが次のように変わります。

```
Disk version
How many files?(0-15)            ・・・・・・・・・・・・③-1
User identifier                  ・・・・・・・・・・・・③-2
EPSON 日本語 BASIC ver X.X
Copyright(c) SEIKO EPSON CORP. 1987, Revised Edition 1988
 XXXXXX Bytes free
```

③-1 同時にオープンする必要のあるファイルの数を入力します。

③-2 User identifier はユーザー識別名と呼ばれるもので、英数字で始まる3文字の文字列を指定します。記憶容量の大きいハードディスクを複数のユーザーが効率良く使用するための識別名になります。ハードディスクに記録するファイルは、この識別名で区別され、識別名が異なるファイルは使用できなくなります。ユーザー識別名が一致したファイルしか使用することはできません。ただし、ユーザー識別名として"999"を指定したときに作成したファイルには、すべてのユーザー識別名から使用できます。

何も入力せずに [↵] キーだけを入力したときは、ユーザー識別名として"□□□"(空白を3つ)を指定したことになります。

またメモリスイッチ SW5-2 を1にすると、ユーザー識別名を問い合せてきません。自動的に"999"を設定したことになります。

ハードディスクからの起動

ハードディスクから起動するためには、あらかじめ BASIC のシステムプログラムをハードディスクに書き込んでおく必要があります。BMENU では「フォーマット」の中の「システムディスク」で領域確保した場合に、システムプログラムが書き込まれます。また「データディスク」を実行した後であれば「ディスクコードコピー」でシステムプログラムを書き込むことができます。

システムプログラムがハードディスクにあると、次の場合に、ハードディスクから BASIC が立ち上がります。

(1) メモリスイッチ SW5-4, -5, -6, -7がすべて0で、フロッピーディスクドライブにシステムディスクをセットしない場合。
BMENU では「立ち上げ装置順位」で「FDD→HDD」に設定した場合

(2) メモリスイッチ SW5-4, -5, -6, -7が0101の場合。このときはシステムディスクがセットされていてもハードディスクから起動します。
BMENU では「立ち上げ装置順位」で「HDD#1」に設定してある場合。

メモリスイッチ SW5-4, -5, -6, -7はシステムを起動するドライブを指定するスイッチです。

1.10.3 40MB ハードディスクの使用方法(拡張フォーマット)

20MB より大きい記憶容量を持つハードディスクに対しては拡張フォーマットを行います。

① BMENU の「ハードディスクメンテナンス」の「全領域初期化」で拡張フォーマットで初期化を行います。このフォーマットを行うとハードディスク上のすべてのデータが消去されますので注意してください。MS-DOS などほかのシステムでフォーマットしてある場合は実行する必要はありません。

② BMENU の「フォーマット」で BASIC の領域を確保します。1MB 単位で空き領域の大きさまで確保できますので、必要な大きさを確保します。また標準フォーマットとは異なり、複数の BASIC 領域を確保することもできます。

③ 次に、この BASIC の領域を「状態変更」で使用状態にします。データの読み書きできる状態を使用状態、データの読み書きのできない状態を休止状態といいます。フォーマットで領域確保しただけでは休止状態のままですのでハードディスクにデータの読み書きすることはできません。「状態変更」により必ず使用状態にします。使用状態になった領域は「*」が表示されます。1台のハードディスクの中で使用状態にできるのは1つだけです。

④ BMENU を終了し、リセットを行い再度 BASIC を立ち上げ直します。この時点からハードディスクが正常に認識され、使用することができます。

ハードディスクからの起動

システムプログラムがハードディスクに書き込まれていて、前ページのメモリスイッチの設定になっていると、ハードディスクからBASICを立ち上げることができます。ただし、拡張フォーマットしたハードディスクからシステムを起動する場合はスタートアッププログラムが起動して、次のような画面を表示します。

```
EPSON PC-286 Series
固定ディスク　スタート・アップ　プログラム　Version X.XX
      Copyright (C) SEIKO EPSON CORP. 1987, Revised Edition 1988

領域NO　　領域名称　　　　　　　　　　　立ち上げ方法選択
   1   自 Disk BASIC X.X    立ち上げ動作  自動立ち上げ解除  次のドライブ
   2      MS-DOS Ver 3.1    立ち上げ動作  自動立ち上げ設定  次のドライブ
   3
   4
   5
   6
   7
   8

矢印キーで領域と立ち上げ方法を選択してください。リターンキーで設定されます。
```

カーソルを起動したいシステム、この場合はBASICですのでBASICに合せます。次にカーソルを「立ち上げ動作」に合せて [↵] キーを押すと、BASICが立ち上がります。

このスタートアッププログラムを実行せずに自動的にBASICを起動したい場合は次のようにします。

カーソルを起動したいシステム、この場合はBASICですのでBASICに合わせます。次にカーソルを「自動立ち上げ解除」に合せて [↵] キーを押します。BASICの欄に「自」が表示されます。さらにカーソルを「立ち上げ動作」に合せていったんBASICを起動します。このあとメモリスイッチSW5-4, -5, -6, -7を0101に変更します。BMENUでは「メモリスイッチ」の「立ち上げ装置順位」で「HDD#1」に設定します。

ディップスイッチSW2-5がONであることを確認して、コンピュータをリセットします。自動的にBASICが立ち上がります。

この設定を変更するには、メモリスイッチSW5-4, -5, -6, -7を0000に変更してスタートアッププログラムを実行するようにします。ディップスイッチSW2-5をOFFにすると、自動的に0000に設定されてスタートアッププログラムを実行します。

1.10.4 リトラクト

ハードディスクが動作中に、電源スイッチを切ったり、強い衝撃を与えると磁気ヘッドがハードディスクに直接接触し、ハードディスクを傷つけることがあります。したがって、ハードディスクを接続しているコンピュータでは電源を切る前に、磁気ヘッドを安全な場所に移動し固定します。これをリ

トラクトあるいはシップディスクなどと呼びます。BASICではCLOSE文を実行することにより、リトラクトが行われます。

例 CLOSE [↵] ハードディスクのアクセスランプが点滅しますので、リトラクトが行われたことを確認できます。

参考 ハードディスクの種類によっては、電源をオフにしたり、長時間データを読み書きしないと自動的にリトラクトするものもあります。このようなハードディスクではCLOSE文を実行してもアクセスランプが点滅しないこともあります。

1.10.5 バックアップとリストア

データのバックアップとは、ハードディスクに記録したデータをほかの記憶装置に移して保管することをいいます。またデータのリストアとはバックアップしたデータをもとどおりハードディスクに記録し直すことをいいます。

誤った操作や故障などにより、ハードディスクに記録したデータが破壊されてしまった場合、あらかじめデータをバックアップしておけば、そのデータをハードディスクにリストアしてデータを復活させることができます。

ファイルのバックアップはBMENUの「ハードディスクメンテナンス」の「ファイルバックアップ」、「ファイルリストア」を使うと便利です。

1.11 オートスタート

オートスタートは、同時にオープンするファイルの数やユーザー識別名を入力せずにBASICを起動する方法です。

オートスタートの設定をすると、なんのキー入力もせずにBASICプログラムを自動的に実行することができます。

オートスタートの設定はユーティリティソフトBMENUで行います。

```
Disk version
How many files?(0-15)………(1)
user identifier?          ………(2)
EPSON 日本語 BASIC ver X. X
Copyright (c) SEIKO EPSON CORP. 1987, Revised Edition 1988
 XXXXXX Bytes free
RUN "GAME"                ………(3)(オートスタートコマンド)
```

オートスタートは上の(1)、(3)をあらかじめシステムディスクのID部に書き込むことによって実現しています。また、オートスタートコマンドはBASIC起動直後に自動的に実行する253文字以内のプログラムです。

(2)のユーザー識別名は、ID部では管理していません。メモリスイッチSW5-2を1にすると、ユーザー識別名に自動的に"999"を設定します。ただしユーザー識別名に"999"以外を設定する場合は次のようにします。

① メモリスイッチSW5-2を1にする。これにより、まずユーザー識別名として自動的に"999"を設定します。BMENUでは「メモリスイッチ」の「固定ディスクユーザー識別名」で「不使用」にします。

② さらにオートスタートコマンドの最初に次のプログラムを付加します。

```
DEF SEG=&h60:POKE &H510, n1:POKE &H511, n2:POKE &H512, n3
```

ここでn1, n2, n3は次の意味です。

n1:ユーザー識別名の先頭の文字の文字コード

n2:ユーザー識別名の2番目の文字の文字コード

n3:ユーザー識別名の3番目の文字の文字コード

例 例えばユーザー識別名として"ABC"を設定する場合は次のようになります。

```
DEF SEG=&h60:POKE &H510, 65:POKE &H511, 66:POKE &H512, 67
```

1.12 メモリスイッチとディップスイッチ

BASIC起動時には、コンピュータ本体のディップスイッチやメモリスイッチの内容を読み込んで、それぞれの設定を行います。このため、ディップスイッチやメモリスイッチの設定を変えた場合は、電源スイッチを一度オフにしてからオンにするか、リセットボタンを押して再度BASICを起動し直すようにしてください。

メモリスイッチについては「付録B メモリスイッチ」を参照してください。ディップスイッチでBASICに関係する主なパラメータは次のとおりです。

ディップスイッチ	機　　能	意　　味	
SW1-4	フロッピーディスクドライブのデバイス名の順番	OFF	本体フロッピーディスクドライブ＃1，＃2 増設フロピーディスクドライブ＃3，＃4
		ON	本体フロッピーディスクドライブ＃3，＃4 増設フロピーディスクドライブ＃1，＃2
SW1-8	グラフィックモードの設定	OFF	8色中8色モードのみ使用(基本グラフィックモード)
		ON	4096色モード使用可能(拡張グラフィックモード)
SW2-3	テキスト画面の文字表示数の設定	OFF	40文字/行
		ON	80文字/行
SW2-4	テキスト画面の文字表示数の設定	OFF	20行/画面
		ON	25行/画面
SW2-5	メモリスイッチの初期化の設定	OFF	電源立ち上げ時にメモリスイッチの値をシステム設定値に設定し直す
		ON	電源立ち上げ時にメモリスイッチの値を変化しない

▒：出荷時設定

第2章

BASICの文法

BASICでプログラムを作成し、さまざまな指示をコンピュータに与えるわけですが、ただやみくもにBASICの命令を並べればよいのではありません。BASICもひとつの言葉である以上、文法があります。

2.1 プログラムの構成

2.1.1 行と行番号

BASICで記述し、特定の手続きや処理をコンピュータが実行する最小単位を行といいます。

プログラムは行の集まりによって構成されます。さらに行は、行番号と文から構成されます。

```
100  PRINT"ABCDE"
行番号    文
```

行番号はプログラムの実行順序を指示するための、1から65529までの整数です。プログラムはGOTO文、GOSUB文、IF～THEN～ELSE文などで実行の順番を指示しない限り、行番号の小さいものから順に実行していきます。

一つの行の長さは行番号を含めて255文字以内です。この範囲内であれば、コロン(：)で区切ることによって、いくつもの文を書くことができます。これをマルチステートメント(複文)といいます。一つの行は[↵]キーを押すことによりメモリにプログラムとして記憶されます。

例
```
100 PRINT "ABCD" : FOR I=1 TO 100 : NEXT
```

文には実行文と非実行文の2種類があります。実行文はBASICが実際に実行する命令を記述したものです。これに対して非実行文はREM文やDATA文などのようにプログラムに注釈を付けたり、READ文で読み出すデータの定義を行います。

2.1.2 ラベル

GOTO文、GOSUB文やIF～THEN～ELSE文などでプログラムの実行順序を変える場合は、直接行番号を指定する代わりにラベルを指定することができます。

行番号は単なる数値の並びであり、行番号自体にプログラムの意味を持たすことはできません。これに対してラベルを用いれば、ラベル自体に意味のある言葉を定義することによりプログラムを見やすいものにすることができます。

ラベル名は分岐先の目印として自由に設定できますが、次の方法に従って付けてください。

(1) ラベル名の先頭には、必ずアスタリスク(＊)をつけてラベルであることを示します。

(2) ラベル名に使用できる文字は英数字、ピリオド(.)であり、英字については大文字、小文字の区別はありません。ただしラベル名の先頭は英文字でなければなりません。

(3) 予約語をラベル名として使用することはできません。

(4) ラベル名の長さは先頭に付け加えるアスタリスク(＊)を含めて1行の範囲内(255文字)です。

(5) ラベルは必ず行の先頭になければなりません。

(6) 同一プログラムの中で同じラベル名を定義することはできません。もし、定義するとDuplicate label(ラベルの二重定義)エラーになります。

例
```
100 ' ラベルを使ったプログラム
110 '
120 *START
130   INPUT "数字を入力してください。";A
140   IF A<100 THEN *MIMAN
150   IF A>=100 THEN *IJOU
160 *MIMAN
170   PRINT A;"は100 未満です。"
180   GOTO *START
190 *IJOU
200   PRINT A;"は100 以上です。"
210   GOTO *START
```

2.2 BASICで使用する文字

2.2.1 文字セット

BASICでは、英文字、数字、カタカナ、グラフィック文字、制御文字(コントロールコード)、ひらがな、漢字を使うことができます。これらは文字セット(キャラクタセット)と呼ばれています。

文字には1バイト文字と2バイト文字があります。1バイト文字は文字コードとして1バイトのコードを持ち、2バイト文字は2バイトのコードを持ちます。

1バイト文字	英文字、数字、記号、カタカナ、グラフィック文字、制御文字
2バイト文字	英文字、数字、カタカナ、ひらがな、漢字、記号など

BASICで用いる予約語、演算子、変数名、数値を表す数字はすべて1バイト文字です。2バイト文字でコマンドを入力するとエラーになります。また2バイト文字の数字で数値定数を表現することはできません。2バイト文字は、文字列とコメントにのみ使用できます。

制御文字はコントロールコードとも呼ばれ､画面上で特別な動作を行わせるのに使用するものです。これらの文字は直接に表示させることはできませんので、CHR$関数などを用いて使用します。

例
```
PRINT CHR$(7)
PRINT CHR$(14)
```

2.2.2 特殊文字

BASIC では次のような文字を特別な機能を持つ文字として使用しています。

文字	機　　能	文字	機　　能
	スペース(空白)	$	文字型宣言子
=	等号(関係演算子)、代入記号(式)	!	単精度実数型宣言子
+	正符号、加算記号、連結記号(文字例)	&	8進、16進表記
−	負符号、減算記号、ハイフン	,	式やデータの区切り
*	乗算記号、ラベルの先頭文字	.	小数点、直前に参照した行の指定
/	除算記号	:	マルチステートメント(複文)
¥	整数除算記号	;	式の区切り
^	べき乗記号	?	PRINT の省略形
(	左かっこ	<	不等号(関係演算子)
)	右かっこ	>	不等号(関係演算子)
%	整数型宣言子	"	文字列指定
#	ファイル番号、倍精度実数型宣言子	'	REM の省略形

2.2.3 予約語

BASIC ではコマンド、ステートメント、関数、演算子など特定目的の使用が決められている単語を予約語としています。あらかじめ使用を決めているのは、それらの単語を変数として使用したときに起きる混乱を避けるためです。したがって、予約語を変数名として使用することはできません。

次にあげる語が BASIC の予約語です。これらを変数名、ラベル名として使用することはできません。また拡張ボード機能により、予約語がふえることがあります。

A ABS
AKCNV$
AND
ASC
ATN
ATTR$
AUTO

B BEEP
BLOAD
BSAVE

C CALL
CDBL
CHAIN
CHR$
CINT
CIRCLE
CLEAR
CLOSE
CLS
CMD
COLOR
COM
COMMON
CONSOLE
CONT
COPY
COS
CSNG
CSRLIN
CVD
CVI
CVS

D DATA
DATE$
DEF
DEFDBL
DEFINT
DEFSNG
DEFSTR
DELETE
DIM
DRAW
DSKF
DSKI$
DSKO$

E EDIT
ELSE
END
EOF
EQV
ERASE
ERL
ERR
ERROR
EXP

F FIELD
FILES
FIX
FN
FOR

FPOS
FRE

G GET
GOTO
GO TO
GOSUB

H HELP
HEX$

I IEEE
IF
IMP
INKEY$
INP
INPUT
INPUT$
INSTR
INT
IRESET
ISET

J JIS$

K KACNV$
KANJI
KEXT$
KEY
KILL
KINPUT
KINSTR
KLEN
KMID$
KNJ$
KPLOAD
KTYPE

L LEFT$
LEN
LET
LFILES
LINE
LIST
LLIST
LOAD
LOC
LOCATE
LOF
LOG
LPOS
LPRINT
LSET

M MAP
MERGE
MID$
MKD$
MKI$
MKS$
MOD
MON

N NAME
NEXT
NEW
NOT

O OCT$
OFF
ON
OPEN
OPTION
OR
OUT

P PAINT
PEEK
PEN
POINT
POKE
POLL
POS
PRESET
PRINT
PSET
PUT

R RANDOMIZE
RBYTE
READ
REM
RENUM
RESTORE
RESUME
RETURN
RIGHT$
RND
ROLL
RSET
RUN

S SAVE
SCREEN
SEARCH
SEG
SET
SGN
SIN
SPACE$
SPC
SQR
SRQ
STATUS
STEP
STOP
STR$
STRING$
SWAP

T TAB
TAN
THEN
TIME$
TO
TROFF
TRON

U USING
USR

V VAL
VARPTR
VIEW

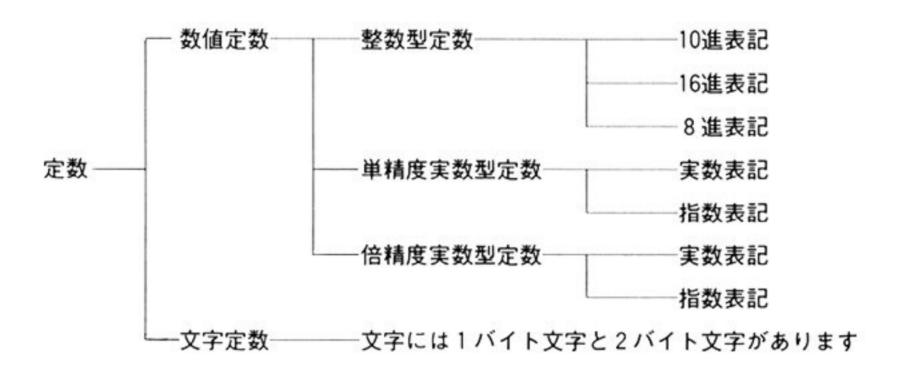
W WAIT
WBYTE
WEND
WHILE
WIDTH
WINDOW
WRITE

X XOR

2.3 定数

BASICで扱うデータには定数と変数の2種類があります。定数とはある値(長さ、重さ、金額など)を具体的な数値または文字列で表したものであり、その値に名前を付けて区別できるようにしたものが変数です。

2.3.1 定数の種類

定数には文字を意味する文字定数と数値を表す数値定数があります。定数はプログラム実行中に値が変わりません。

2.3.2 数値定数

数値定数はそれ自身で定まった値を表すもので、整数型と実数型に分けられます。実数型は数値の精度の違いにより、単精度実数型と倍精度実数型に分けられます。数値定数は正あるいは負の数または0です。負の数の前には必ず負符号(−)を付けますが、正の数の場合の正符号(+)は省略できます。また数字と数字の間の空白(スペース)は無視されます。

(1) **整数型定数**

整数型で扱う数値は−32768から+32767までの整数です。整数の表現は次のいずれかの形式になります。

① **10進表記**

−32768から32767までのすべての整数を指定できます。

例
```
3860%
-585%
6856%
```

② **16進表記**

先頭に &H を付けた4桁までの16進数字(0、1、2、3、4、5、6、7、8、9、A、B、C、D、E、F)で数値を表現します。

例

&H26　10進表記では38 $(2\times16+6)$

&H43B　10進表記では1083 $(4\times16^2+3\times16+11)$

③　8進表記

先頭に&O(英文字のオー)または&を付けた6桁までの8進数字(0、1、2、3、4、5、6、7)で数値を表現します。

例
&26　　10進表記では22(2×8+6)
&O432　10進表記では282(4×8^2+3×8+2)

プログラム中で数値を8進、16進で表現しても実行結果はすべて10進数で出力されます。

例
```
PRINT  &H45+&H10
 85
```
10進表記

(2)　実数型定数

実数型定数は単精度実数型定数と倍精度実数型定数に分けられます。

①　単精度実数型定数

単精度実数型定数は有効数字約7桁の精度で格納し、結果は7桁目を四捨五入し6桁の10進数で出力します。プログラム中で次のような表現をすると単精度実数型定数とみなされます。

表現方法	例
仮数部が7桁以下の数	97.5
最後に感嘆符(!)を付けた数	12300000!

単精度実数で扱える数値は－1.70141E＋38～＋1.70141E＋38までです。

②　倍精度実数型定数

倍精度実数型定数は有効数字約17桁の精度で格納し、結果は17桁目を四捨五入し16桁の10進数で出力します。プログラム中で次のような表現をすると倍精度実数型定数とみなされます。

表現方法	例
仮数部が8桁以上の数	12300000.5
Dを用いた指数形式	1.23D＋10(1.23×10^{10})
最後に#を付けた数	12300#

倍精度実数で扱える数値は－1.701411834604692D＋38～＋1.701411834604692D＋38までです。

2.3.3　文字定数

文字定数は文字列を引用符(")で囲んで表します。文字列の長さは255文字までです。特に文字列の長さが0の場合を、空文字列(ヌルストリング)といい""で表します。

注意　ここで言う文字数は1バイト文字を1文字としたものです。したがって2バイト文字の場合は2文字分必要になるとともに、前後のKI/KOコードの文字数も含まれます。

文字列の途中に引用符を使用する場合はCHR$関数と文字列の連結記号(+)を用います。

例
```
A$=""                                  (ヌルストリングの代入)
PRINT "A"+CHR$(34)+"B"                 (A"Bと表示される)
INPUT  "数字を入力してください" ; N
```

2.4 変数

変数は値を一時的に蓄えるものであり、数値型変数と文字型変数の2種類があります。変数に代入するデータ(文字列、数値)と変数の型は一致していなければなりません。

2.4.1 変数名

変数には英字で始まる40文字以下の英数字を組み合わせた名前(変数名)を付けます。英字については大文字、小文字の区別はしません。小文字で入力しても自動的に大文字に変換します。

また予約語を変数名として使用することはできません。予約語を変数名として使用するとエラーが発生し、プログラムの実行は中断します。ただし予約語を変数名の一部として含む場合は問題ありません。またFNを変数名の最初の2文字として使用することはできません。FNはユーザー定義関数名とみなされます。

2.4.2 変数の型

変数も定数と同じように次のような型があり、これらは変数名の後ろにつけた型宣言文字により区別されます。

	型宣言文字	変数の型	内部表現
文字変数	$	文字型	最大255文字
数値変数	%	整数型	1変数につき2バイト
	!	単精度実数型	1変数につき4バイト
	#	倍精度実数型	1変数につき8バイト

例
```
A$
NUM%
MAX!
MIN
LIMIT#
```

型宣言文字を省略した場合は、通常、単精度実数型とみなします。

このほかに変数の型を宣言する方法として、次の命令を使う方法があります。

	型の宣言	変数の型
文字変数	DEFSTR	文字型
数値変数	DEFINT	整数型
	DEFSNG	単精度実数型
	DEFDBL	倍精度実数型

ただし、型宣言文字で指定する方法が優先されます。

例
```
100    DEFSTR A−C  変数名の先頭がA, B, Cで始まる変数の型を文字型とする
110    AX%=10      ただしAX%は整数型変数
```

同じ変数名でも型が異なれば、別の変数として扱われます。

2.4.3 配列変数

配列は同じ型を持つ複数個の変数の集まりです。変数を配列変数として宣言するにはDIM文を用います。配列に含まれる各々の値を要素といい、それぞれの要素は添字によって区別します。配列変数にも文字型、整数型、単精度実数型、倍精度実数型の4つの型があります。

例
```
DIM DAY(31)
```
単精度実数型の配列変数で要素の数は32(添字は0から31)になります

DAY(0)	DAY(1)	DAY(2)	DAY(3)	DAY(4)	…	DAY(31)

添字の下限を0にするか1にするかはOPTION BASE文で設定できます。BAISC起動時の設定は0になっています。

DIM文で配列変数の宣言をしなくても配列を使用することができます。ただし、この場合の配列の添字の上限は10です。

添字の個数により配列の次元数が設定されます。配列の次元数は最大255まで可能です。

例
```
DIM C(20, 20)… 2次元配列
```

C(0, 0)	C(0, 1)			
C(1, 0)	C(1, 1)			
			C(18, 19)	C(18, 20)
			C(19, 19)	C(19, 20)
			C(20, 19)	C(20, 20)

2.5 型の変換

BASICは必要に応じて自動的に数値データの型の変換を行います。ただし文字型と数値型との間の変換はVAL関数やSTR$関数などを用いないかぎり行うことはできません。型の変換は次のような場合に行われます。

(1) 数値定数や数値変数の値を異なる型の数値変数に代入する場合に、代入される数値変数の型に変換して代入します。

- 実数型定数を整数型変数に代入する場合に、小数点以下を四捨五入します。変換した数値が整数型で扱う数値の範囲を超えている場合は、Overflow(桁あふれ)エラーになります。

例
```
10  A%=23.545
20  PRINT A%
RUN
 24
```
自動的に小数点以下を四捨五入します。

例
```
10  A%=32789
20  PRINT A%
RUN
Overflow in 10
```
整数型で扱う数値の範囲は－32768から＋32767までのため桁あふれが発生します。

• 倍精度実数値を単精度型変数に代入する場合に、倍精度実数値を単精度実数で表すことのできる精度に四捨五入して代入します。

例
```
10  A!=12.3456789#
20  PRINT A!
RUN
 12.3457
```

• 単精度実数値を倍精度実数型変数に代入した場合に、表示したときの値が異なることがありますので注意してください。

例
```
10  A#=0.1!
20  PRINT A#
RUN
 .1000000014901161
```
```
10  A#=0.1#
20  PRINT A#
RUN
 .1
```

(2) 精度の異なる数値間で算術演算や関係演算を行う場合は、低い精度の数値を高い精度の数値に変換してから、高い精度で演算を行います。

例
```
10  A#=7#/6
20  PRINT A#
RUN
 1.166666666666667
```

7#/6の除算を7#/6#のような倍精度の除算として実行します。

例
```
10  A=7#/6#
20  PRINT A
RUN
 1.16667
```

(3) 論理演算や¥、MODを行う場合は、扱う数値はすべて整数に変換してから演算を行います。その結果も整数値で与えます。整数値の範囲は−32768から32767までで、この範囲を超える数値を指定した場合は、Overflow(桁あふれ)エラーになります。

例
```
PRINT 1.56 OR 1.2        (2 OR 1と同じ)
 3
PRINT 1.46 OR 1.2        (1 OR 1と同じ)
 1
```

2.6 式と演算子

式とは、定数や変数を演算子で結び付けたものや定数だけ、変数だけのものをいいます。BASICには次の5つの式があります。

算術演算式
関係演算式
論理演算式
関数
文字式

2.6.1 算術演算式

算術演算式は算術演算子を用いて数値データの計算を行うものをいいます。

算術演算子	演算内容	書式例	数学的表現
+	加算	A+B	A+B
−	減算	A−B	A−B
*	乗算	A*B	A×B
／	除算	A／B	A÷B
+, −	符号	−A	−A
^	べき乗	A^B	A^B
¥	整数の除算	A¥B	—
MOD	整数の剰余	A MOD B	—

例	BASICでの表現	数学的表現
	3*X+2*Y	3X+2Y
	A−5*B	A−5B
	A+B/C	A+B÷C
	X^2+2*X+1	X^2+2X+1
	X*(−Y)	X・(−Y)

(1) 演算中に0で除算を行ったり、0に負の数のべき乗を実行した場合は/0(0で除算)を表示します。ただし、演算結果は扱う範囲内の最大値となり、そのまま演算を続けます。

(2) 整数の除算は"¥"記号で表されます。扱われる数値の実数の場合は、演算が実行される前に小数第一位で四捨五入されます。商は小数点以下が切り捨てられた整数となります。

例		
	10¥3=3	(10/3=3.3333……)
	21.75¥3=7	(22/3=7.333……)

(3) 整数の剰余は、"MOD"によって行われ、扱われる数値が実数の場合は、演算が実行される前に小数点以下が四捨五入されます。結果は整数の割り算の余りです。

例		
	10 MOD 3=1	(10/3=3　余り1)
	21.75 MOD 3=1	(22/3=7　余り1)

(4) 演算の結果がその型で表現することのできる範囲を越えた場合は、オーバーフロー(桁あふれ)エラーが発生します。演算の種類によって、その後の実行を続けたり中断したりします。継続する場合は、OV(Overflow の略)を表示します。

2.6.2 関係演算式

関係演算式は次のような関係演算子を用いて2つのデータを比較するものをいいます。比較の際に2つのデータは両方とも数値型あるいは文字型でなければなりません。比較の結果、正しければ−1(真)、誤っていれば0(偽)を返します。

関係演算子	書式	内容
=	A=B	AとBが等しい
<>または><	A<>BまたはA><B	AとBが等しくない
<	A<B	AがBより小さい
>	A>B	AがBより大きい
<=または=<	A<=BまたはA=<B	AがBより小さいかまたは等しい
>=または=>	A>=BまたはA=>B	AがBより大きいかまたは等しい

関係演算はIF〜THEN〜ELSE文でプログラムの流れを変える場合などに使用します。

(1) 数値の比較

算術演算と関係演算が1つの式に組み合わせられているときには、算術演算を先に行います。

例 A+B>(X−1)*Y

A+Bの値が(X−1)*Yより大きいときに−1(真)に、小さいか等しいときに0(偽)になります。

(2) 文字列の比較

文字列の比較は、各文字列の先頭から1文字ずつ文字コードの比較を行います。すべての文字コードが等しい場合、2つの文字列は等しくなります。異なった文字コードが見つかると、そこで演算を中止し、その文字コードの大きい文字列を大きい文字列とします。また一方の文字列の長さが短く、それまでに比較した部分が等しい場合は、長い文字列を大きい文字列とします。

例

"FILENAME"="FILENAME"　真
"AA"<"AB"　真(Bの文字コードはAの文字コードより大きい)
"ABCDE"<"ABCDEF"　真

2.6.3 論理演算式

論理演算式は次のような論理演算子を用いて論理演算を行うものをいいます。論理演算子の対象となる数値は−32768から32767までの整数に変換します。この範囲内にない場合はOverflow(桁あふれ)エラーになります。整数値は内部では16ビットの2進数で表しており、論理演算は2つの数値の対応するビットごとに指定した論理演算を実行します。

論理演算子の種類と、それらによって求められる結果の内容を次に示します。

NOT(否定)

A	NOT A
1	0
0	1

AND(論理積)

A	B	A AND B
0	0	0
0	1	0
1	0	0
1	1	1

OR(論理和)

A	B	A OR B
0	0	0
0	1	1
1	0	1
1	1	1

XOR(排他的論理和)

A	B	A XOR B
0	0	0
0	1	1
1	0	1
1	1	0

IMP(包含)

A	B	A IMP B
0	0	1
0	1	1
1	0	0
1	1	1

EQV(同値)

A	B	A EQV B
0	0	1
0	1	0
1	0	0
1	1	1

例

```
PRINT   127 AND 16
 16
OK
```

127は2進数では0000000001111111
16は2進数では 0000000000010000
結果は16 0000000000010000

各ビットごとに論理積(AND)を計算すると、その結果は16になります。

負の数は2の補数で表現します。これは数値の絶対値の各ビットの1と0をそれぞれ反転させ、1を加えたものです。

例

```
-16
```

16は2進数では0000000000010000
したがって 1111111111101111(ビットの反転)
−16の内部表現1111111111110000(1を加える)

2.6.4 関数

関数は関数の対象となる1つ以上のデータについて、BASIC上であらかじめ定義した演算を行うものです。

数値関数としてはSIN(正弦)、SQR(平方根)などを用意しています。また文字関数としては、ある文字列の一部を取り出すMID$や文字の文字コードを与えるCHR$関数などを用意しています。関数の種類は「日本語Disk BASICリファレンスマニュアル」を参照してください。

例
```
PRINT SQR(2)      √2を計算する。
PRINT ASC("A")   "A"の文字コードを求める
```

定義されていない関数についてはプログラム中でDEF FN文を用いて定義することができます。

例
```
100 DEF FNLOG10(X)=LOG(X)/LOG(10#)   常用対数の定義
```

2.6.5 文字式

文字式とは、文字変数や文字定数を演算子により結合した式のことです。また単に一つの文字変数および文字定数であっても文字式といいます。文字列に対しては関係演算や文字関数のほか次の演算を行うことができます。

文字列の連結

加算記号(+)を使用して文字列の連結を行うことができます。この演算子は算術演算での演算子と異なり、2つの文字列の加算ではなく、2つの文字列の連結を意味します。

例
```
PRINT "123"+"456"
123456             123456を表示
```

例
```
10  DRIVE$="1":FLNAME$="TEST"
20  FDSC$=DRIVE$+":"+FLNAME$
30  PRINT  FDSC$
```

2.6.6 演算子の優先順位

式の中に複数の演算子が用いられている場合は優先順位の高いものから実行します。また同じ順位の演算子の場合は式の左から右への順に実行します。

かっこ"( )"で囲まれた式および関数は最も優先的に実行します。

実行の順序

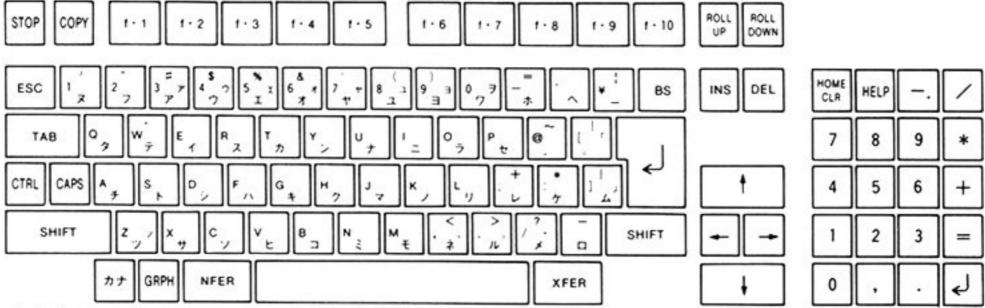
① ( )で囲まれた式
② 数値関数・文字関数
③ べき乗(^)
④ マイナス符号(−)
⑤ 乗算・除算(*,/)
⑥ 整数の除算(¥)
⑦ 除算の剰余(MOD)
⑧ 加算・減算(+,−)
⑨ 関係演算(<,>,=など)
⑩ 否定(NOT)
⑪ 論理積(AND)
⑫ 論理和(OR)
⑬ 排他的論理和(XOR)
⑭ 包含(IMP)
⑮ 同値(EQV)

第3章

キーボードとスクリーンエディタ

BASICに対する命令や指示は、キーボードから特定の意味を持つ文字(単語)を打ち込んで行います。ここではBASICを利用する上で必要不可欠なキーボードの使い方をまとめて説明します。

3.1 キーボードの配列とキー入力

キーボードは次のような機能別にまとめた101個のキーで構成されており、キーの配列はJISに準拠した配列になっています。PC-286L/LE/LSシリーズでは84キーですが NUM キーにより同等のキーの使用が可能です。

キー入力には、次の4つのモードがあります。

- 英数字入力モード
- カタカナ入力モード
- グラフィック入力モード
- 日本語入力モード

3.1.1 英数字入力モード

英数字を入力するモードで、 カナ キーがロック*していない状態です。

このモードではキートップの左と上に書かれているアルファベット、数字、英記号を入力することができます。

- CAPS キーがロック*していない場合

  アルファベットの小文字を入力します。 SHIFT キーを押し下げたままキーを押すと大文字を入力できます。

- CAPS キーがロックしている場合

  アルファベットの大文字を入力します。 SHIFT キーを押し下げたままキーを押すと小文字を入力できます。

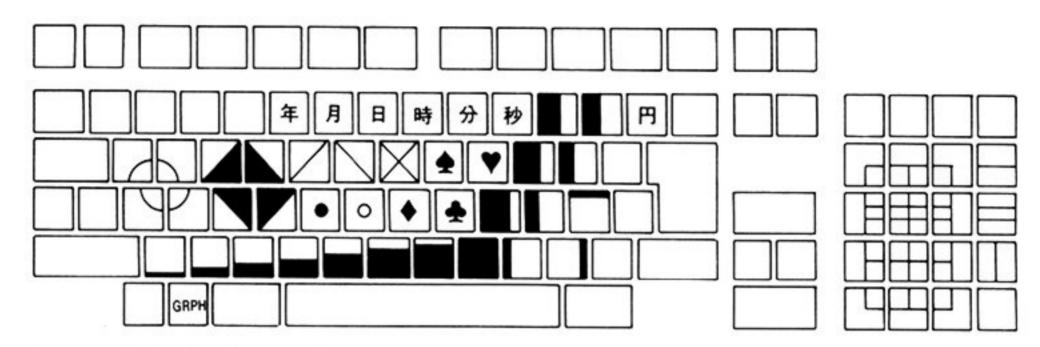
  キートップの上側の英記号は、 CAPS キーの状態に係わらず、 SHIFT キーを押し下げたままキーを押すと入力できます。

*PC-286/286X/386, PC-286U/USシリーズでは カナ キーあるいは CAPS キーのLEDが点灯している状態，その他の機種では カナ キーあるいは CAPS キーが半分押し下げられた状態をいいます。

3.1.2 カタカナ入力モード

カナ キーを押して、ロックした状態になるとカタカナモードになります。そのままキーを押すとキートップの下に書かれているカタカナが入力されます。SHIFT キーを押し下げたまま、キーを押すとキーをキートップ右のカタカナとカナ記号が入力されます。

3.1.3 グラフィック入力モード

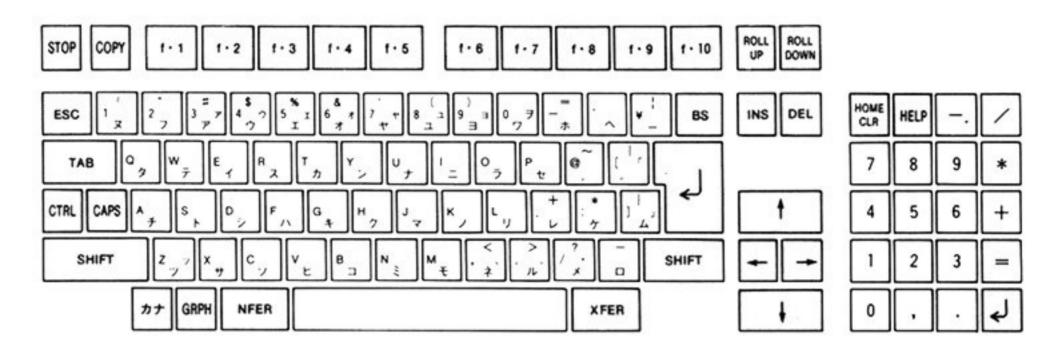
GRPH キーを押し下げたままキーを押すとグラフィック文字を入力できます。このモードをグラフィック入力モードと呼びます。各キーで入力できるグラフィク文字は次のとおりです。

3.1.4 日本語入力モード

CTRL キーを押しながら XFER キーを押すと日本語入力モードになります。このモードでは漢字やひらがななどの2バイト文字の入力ができます。日本語の入力については、かな漢字変換方式とコード入力方式が使用できます。

かな漢字変換方式	日本語入力機能を読み込んでいるときに使用できます。初期設定はこの状態になっています。日本語入力の詳細については「第4章　日本語入力」を参照してください。
コード入力方式	日本語入力機能を読み込んでいないときに使用します。JIS漢字コードを4桁入力すると、対応する1文字を入力することができます。このとき、かな漢字変換方式のように画面の表示が変わりませんので注意してください。日本語入力機能はユーティリティプログラムBMENUにより切り離すことができます。JIS漢字コードについてはコンピュータ本体のユーザーズマニュアルを参照してください。

日本語入力モードからは、再度 CTRL キーを押しながら XFER キーを押すと、もとのモードに戻ります。

3.1.5 特殊キー

下図に示したキーは、文字の入力でない特殊な機能を持っています。

[STOP] (ストップキー)

プログラムの実行を停止させます。

[COPY] (コピーキー)

画面のハードコピーをプリンタに出力します。「7.2 画面ハードコピー」を参照してください。

[f・1] ～ [f・10] (ファンクションキー)

キーボードから1文字1文字入力する手間が省けるように、よく利用するBASICの命令などをファンクションキーに設定することができます。これらはKEY文で自由に変えることができます。BASIC立ち上げ時の設定は次のようになっています。

f・1	load "	f・6	save "
f・2	auto	f・7	key
f・3	go to	f・8	print
f・4	list	f・9	edit . [↵]
f・5	run [↵]	f・10	cont [↵]

ファンクションキーを押すと、それぞれに割り当てられている文字列が入力されます。各ファンクションキーには最大15文字の文字列を割り当てることができます。文字列を割り当てるにはKEY文を使います。

例 [f・4] に「RUN [↵]」を設定する

```
KEY 4, "RUN"+CHR$(13)
```

また設定した内容は、CONSOLE文で画面の最下行に表示させることができます。

[ROLL UP] (ロールアップ) [ROLL DOWN] (ロールダウン)

エディタモードで画面をスクロールするのに使います。「3.2 スクリーンエディタ」を参照してください。

[ESC] (エスケイプキー)

日本語入力や、ユーティリティ実行中にキャンセルキーとして使用します。

[BS] (バックスペース)

カーソル位置の左の1文字を消去し、カーソル位置から右側の文字列を左に詰めます。

[INS] (インサートキー)

キー入力を挿入(インサート)モードにします。挿入モードでは、カーソル位置にキー入力した文字を挿入し、カーソル以降の文字列を順次右にずらします。

挿入モードからは、以下のキーによって上書き(オーバーライト)モードに切り替わります。

- もう一度 [INS] キーを押す
- カーソル移動キー([↑] [↓] [→] [←])
- [↵] キー

[DEL] (デリート)

カーソル位置の左の1文字を消去し、カーソル位置から右側の文字列を左に詰めます。

[HOME CLR] (ホーム/クリア)

テキスト画面のスクロール領域のデータを消去(クリア)し、カーソルをホームポジション(スクロール領域の左上隅の位置)へ移動します。

[SHIFT] キーを押し下げながら入力するとカーソルだけがホームポジションへ動きます。

[HELP] (ヘルプ)

- プログラム実行中にエラーが発生したとき、エラーの発生した行の内容と位置を表示します。
- ON HELP GOSUB 文による割り込みを発生します。

[TAB] (タブ)

カーソルを8桁づつ右へ動かします。同時にカーソルからタブ位置までを消去します。

[↵] (リターン)

1行の区切りを示し、カーソルを次の行の先頭に動かします。

[CTRL] (コントロール)

以下に示すように、他のキーと組み合わせて使用します。[CTRL] + [A] は [CTRL] キーを押し下げたまま [A] キーを押すことを意味します。カタカナ入力モードでも使用できます。

[CTRL] + [A]	[HELP] キーと同じ働きをします。
[CTRL] + [B]	カーソルを前の単語の先頭に動かします。
[CTRL] + [C]	プログラムの実行を中断し、BASIC コマンドモードに戻します。 [STOP] キーと同じ働きです。 なお、このキーはプログラム実行時のみ有効です。
[CTRL] + [D]	カーソルの位置している単語のカーソルから右を消します。
[CTRL] + [E]	現在のカーソル位置からその行の終わりまでを消去します。
[CTRL] + [F]	カーソルを、次の単語の先頭に動かします。
[CTRL] + [G]	ブザーを鳴らします。
[CTRL] + [H]	カーソル位置の左の1文字を消します。 [BS] キーと同じ働きです。

CTRL + I	カーソルを次のタブ位置まで動かします。同時にカーソルからタブ位置までを消します。TAB キーと同じです。
CTRL + J	LF コードをカーソル位置に挿入します。挿入モードであれば、行を分割し2行にします。
CTRL + K	カーソルをホームポジションへ動かします。SHIFT + HOME CLR と同じ働きです。
CTRL + L	テキスト画面のスクロール領域を消去します。HOME CLR キーと同じ働きです。
CTRL + M	⏎ キーと同じ働きをします。
CTRL + N	カーソルを、次の単語の先頭へ移動します。
CTRL + O	テキスト画面への表示だけを停止します。もう一度押すと再開します。CTRL + S と異なり画面への表示だけを停止するだけですので停止してから、再開するまでに表示するはずだったデータは失われます。なお、このキーはプログラム実行時のみ有効です。
CTRL + Q	カーソルを、カーソルの位置する論理行の先頭の文字に移動します。
CTRL + R	挿入モードにします。INS キーと同じ働きです。
CTRL + S	プログラムの実行を一時停止します。また LIST、LLIST コマンドの実行を一時停止します。STOP, CTRL + C, COPY, SHIFT, カナ キー以外のキーを押せば、実行が再開されます。なお、このキーはプログラム実行時のみ有効です。
CTRL + U	カーソルのある行を削除します。
CTRL + X	カーソルのある行の最後の文字の、右側にカーソルを動かします。
CTRL + Z	カーソルの位置から画面の終わりまでを消去します。
CTRL + XFER	日本語入力モードを変更します。

CAPS (キャピタルロック)

このキーを一度押すとロックし、もう一度押すとロックが解除されます。

このキーのロックした状態でアルファベットキーを押すと大文字が入力されます。小文字を入力する場合は SHIFT キーを押し下げたまま、そのキーを押します。「3.1.1 英数字入力モード」を参照してください。

↑ ↓ → ← (カーソル移動キー)

カーソルを矢印の向きに動かします。

SHIFT (シフト)

キートップの上や右に書かれた英記号やカナ記号を入力する際に使います。

カナ (カナ)

このキーを一度押すとロックし、もう一度押すとロックが解除されます。

カタカナモードにします。「3.1.2 カタカナ入力モード」を参照してください。

GRPH (グラフィック)

グラフィック文字の入力に使います。「3.1.3　グラフィック入力モード」を参照してください。

XFER (変換) NFER (非変換)

日本語入力モードで使います。「3.1.4　日本語入力モード」を参照してください。

3.2 スクリーンエディタ

2～3行程度のプログラムならば良いのですが、プログラムが何十行にもおよぶと、入力ミスも起きますし、プログラムの編集や修正の必要もでてきます。プログラムをいちいち書き直していたのでは大変です。BASICの持っているスクリーンエディタ機能により、カーソルを自由に移動させて、プログラムの編集と修正を行うことができます。

3.2.1 プログラムの編集と修正

プログラムの入力

BASICを起動して、画面に"OK"という文字が表示されていれば、プログラムの入力が可能です。スクリーンエディタを用いてプログラムの編集作業の一例を示します。

① 画面をきれいにします。表示内容を消去し、カーソルを画面左上に移動します。そのためには HOME/CLR キーを使います。

参考　グラフィック画面に図形などを表示させている場合は、HOME/CLR キーでは消去できません。キーボードからCLS 3 ↵ と入力します。

② CAPS キーと カナ キーがロックされていないことを確認し、次のようにキーボードを打ってみます。

10　print time$ ↵　　　時間を画面に表示するプログラムです。

注意　プログラムを1行入力したら、必ず ↵ (リターン)キーを押します。カーソルは次の行の先頭に移動します。↵ キーは、その行をコンピュータのメモリに記憶させる重要な働きを持っています。したがって、プログラムを修正したあとなども、必ず ↵ キーを押すようにしてください。この場合、カーソルはその行の中であれば、どこにあっても構いません。

プログラムの実行

③ RUN ↵ と入力すると、プログラムは実行され、次の行に現在の時間が表示されるはずです。

文字の追加

④ コンピュータに入っているプログラムの内容を見たいときには、LIST ↵ と入力します。すると、先ほど入力したプログラムが表示されます。ただし、先ほどとは異なり、すべて大文字で表示されたはずです。このようにBASICでは小文字で入力したプログラムも、自動的に大文字に変換されます。ただし、二重引用符(")で囲まれた文字列や、DATA文で定義した文字定数などは変換されません。

```
list [↵]
10  PRINT TIME$
OK
```

⑤ 時間だけでなく日付も表示されるように修正します。カーソルをTのところに持っていき、[INS]キーを押します。

```
10  PRINT [T]IME$
```

挿入モードになりますので、DATE$; "　" ; と入力します。

```
10  PRINT date$ ; "  " ; [T]IME$
```

[↵]キーを押すのを忘れないようにしてください。

行の追加

⑥ 日付と時間を、画面の右上に表示するようにプログラムを追加します。

```
5  locate 0,63 [↵]
list [↵]
5  LOCATE 0,63
10  PRINT DATE$ ; "  " ; TIME$
```

このように、後から入力した行でも、行番号の小さい順に並び変えられます。

このプログラムを実行すると、予想と反して画面左下に日付と時刻が表示されます。

文字の修正

⑦ カーソルを5行の0のところに持っていき、[INS]キーを押します。

```
5  LOCATE [0], 63
10  PRINT DATE$ ; "  " ; TIME$
```

63と入力します。

```
5  LOCATE 63[0], 63
```

⑧ [→]キーでカーソルを，に持っていきます。これで挿入モードから抜け出します。

```
5  LOCATE 630[,]63
```

⑨ [DEL]キーを押すと、カーソルの前の文字0が消去され、カーソルの後ろの文字列が左に詰められます。

```
5  LOCATE 63[,]63
```

⑩ 同じように後ろの63を0に修正します。

```
5  LOCATE 63, 0
```

最後に [⏎] キーを押すのを忘れないようにしてください。

⑪ このままですと、一度日付と時間を表示してプログラムを終了していまいます。いつも正確な時間を表示するようにプログラムを修正します。20 GOTO 5 [⏎] と入力します。

```
5  LOCATE 63, 0
10 PRINT DATE$ ; "  " ; TIME$
20 gotu 5
```

└Oを入力すべきところをUを入力している。

このプログラムを実行すると20行にエラーがありますので、プログラムの実行が中断されます。

エラーの修正

プログラムにエラーがあると、エラーのある行で実行が中断され、エラーの種類と、エラーの起きた行の行番号を画面に表示します。上の例ですと次のようなメッセージが表示されるはずです。

```
Syntax error in 20
```

エラーの種類によっては、エラーのある行が表示され、エラーのある単語か、その次の単語の先頭でカーソルが点滅します。この場合はスクリーンエディタを使ってプログラムを修正します。
[HELP] キーを押すと、エラーメッセージで指定された行が表示されます。カーソルはエラーのある単語か、その次の単語の先頭でカーソルが点滅します。

[HELP]

```
20 GOTU 5
```

この場合は、カーソルの直前のGOTUの中にエラーがあります。GOTUのUをOに修正して [⏎] キーを押します。

参考 EDITコマンドを使うと、指定した行を画面の最上行に表示して、プログラムを修正することができます。(エディットモード)
EDIT 行番号 [⏎]
[ROLL UP] キーと [ROLL DOWN] キーで画面をスクロールすることができます。

プログラムの編集

⑫ プログラムの行番号を揃えるためにRENUMコマンドを実行してみます。

```
renum  100 [⏎]
list [⏎]
100 LOCATE 63, 0
110 PRINT DATE$ ; "  " ; TIME$
120 GOTO 100
OK
```

このようにRENUMコマンドを使うと行番号を分かりやすくすることができます。

行の削除

⑬ プログラムの中から簡単に指定した行を削除することもできます。
120行のGの前にカーソルを持っていきます。

```
120□GOTO 100
```

［CTRL］キーを押し下げたまま［E］キーを押します。すると、カーソルの右側の文字列が消去されます。このまま［↵］キーを押します。

```
120
```

LISTコマンドを実行すると120行が削除されています。

```
list ［↵］
100  LOCATE 63, 0
110  PRINT DATE$ ; "  " ; TIME$
OK
```

参考 "OK"と表示されている下の行で120［↵］と入力しても、同じように120行を削除することができます。

3.2.2 カーソルの移動

HOME CLR 、 CTRL + L	画面の表示内容を消去し、カーソルを左上隅(ホームポジション)に戻します
SHIFT + HOME CLR 、 CTRL + K	カーソルをホームポジションに戻します。
→	カーソルを右に移動します。
←	カーソルを左に移動します。
↑	カーソルを上に移動します。
↓	カーソルを下に移動します。
CTRL + F 、 CTRL + N	カーソルを次の単語の先頭へ移動します。単語の区切りは空白やカンマ(,)、ピリオド(.)、ハイフン(－)です。
CTRL + B	カーソルを前の単語の先頭へ移動します。
TAB 、 CTRL + I	カーソルを次のタブ位置に移動します。(タブ位置は画面上に1バイト文字8文字ごとに設定されている位置のこと)移動した部分はスペースで消去されます。
CTRL + Q	カーソルを、カーソルの位置する論理行(↵ キーによって入力したプログラム行)の先頭の文字に移動します。
CTRL + X	カーソルを、カーソルの位置する論理行(↵ キーによって入力したプログラム行)の最後の文字に移動します。
STOP 、 CTRL + C	カーソルを、カーソルの位置する論理行の次の行の先頭の文字に移動します。

SHIFT + HOME CLR でホームポジションへ

```
560   PX=FIX(PX/2)
570   IF PX>7 THEN BIT=23-PX ELSE BIT=7-PX
580   BPT=VAL("&H"+HEX$(2^BIT))
590   CD=KP%(PY+2) AND BPT
600   IF CD=0 THEN *KPSET ELSE *KPRESET
610 '
620 *SELECT
630   IF PY=3 THEN GOSUB *KPCLS ELSE *KPEND
640   GOTO *EXIT
650 *KPSET
660   COLOR 1 : LOCATE PX*2,PY : PRINT "■";
670   KP%(PY+2)=KP%(PY+2) OR BPT
680   BEEP : GOTO *EXIT
690 *KPRESET
700   COLOR 7 : LOCATE PX*2,PY : PRINT "■";
710   KP%(PY+2)=KP%(PY+2) AND (NOT BPT)
720   BEEP : GOTO *EXIT
730 *EXIT
```

CTRL + F で次の単語へ

CTRL + B で前の単語へ

CTRL + Q で行の先頭へ

CTRL + X で行の終わりへ

STOP で次の行へ

3.2.3 消去/削除と挿入

キー	機能
CTRL + D	カーソルのある単語の、カーソルから右側を消去します。
CTRL + E	カーソルの位置から、その行(論理行)の終わりまでを消去します。
CTRL + J	カーソル位置にLFコードを挿入します。挿入モードであれば行を分割します。
BS , DEL , CTRL + H	カーソルの直前の一文字を消去し、カーソルの右側の文字列を左に詰めます。
INS , CTRL + R	挿入(インサート)モードにします。挿入モードになると、入力した文字をカーソル位置に挿入していきます。挿入モードから抜け出すには、もう一度 INS (CTRL + R) キーを押すか、カーソル移動キーを押します。
CTRL + U	カーソルの位置する行を消去します。
CTRL + Z	カーソルの位置から画面の終わりまでを消去します。

```
560    PX=FIX(PX/2)
570    IF PX>7 THEN BIT=23-PX ELSE BIT=7-PX
580    BPT=VAL("&H"+HEX$(2^BIT))
590    CD=KP%(PY+2) AND BPT
600    IF CD=0 THEN *KPSET ELSE *KPRESET
610 '
620 *SELECT
630    IF PY=3 THEN GOSUB *KPCLS ELSE *KPEND
640    GOTO *EXIT
650 *KPSET
660    COLOR 1 : LOCATE PX*2,PY : PRINT "■";
670    KP%(PY+2)=KP%(PY+2) OR BPT
680    BEEP : GOTO *EXIT
690 *KPRESET
700    COLOR 7 : LOCATE PX*2,PY : PRINT "■";
710    KP%(PY+2)=KP%(PY+2) AND (NOT BPT)
720    BEEP : GOTO *EXIT
730 *EXIT
740    FOR I=1 TO 200 : NEXT
750    PEN ON : KPLOAD KC,KP% : RETURN  *PENLOOP
760 '
770 *KPEND
780  COLOR 7 : CLS 3 : PEN OFF : END
OK
```

CTRL + D で単語削除

CTRL + E で行の終わりまで削除

CTRL + U でカーソルの位置する行を削除

CTRL + Z で画面の終わりまで削除

第4章

日本語入力

漢字やひらがななどの2バイト文字を入力する方法には、かな漢字変換方式とコード入力方式があります。日本語入力を利用すれば、日本語でメッセージを表示したり、データとして日本語の文字列を入力することができます。

かな漢字変換方式には、EGBridgeを採用しています。

EGBridgeは株式会社エルゴソフトの登録商標です。

4.1 日本語入力モード

CTRL キーを押し下げたまま XFER キーを押したり、プログラム中でKINPUT文を実行したときに日本語入力モードになります。このモードでは漢字やひらがななどの2バイト文字の入力を行うことができます。

日本語入力には、BASIC起動時の設定により、かな漢字変換とコード入力の2つの入力方法があります。

入力方法	機能
かな漢字変換方式	ひらがなで入力した文字を漢字に変換して入力する方法です。かな漢字変換のためのプログラムと辞書ファイルが必要になります。初期設定はこの状態になっています。
コード入力方式	直接、文字のJIS漢字コードを入力して日本語を入力する方法です。かな漢字変換プログラムや辞書ファイルは必要ありません。BASICユーティリティ「BMENU」によって「日本語入力機能メンテナンス」の「環境設定」で「かな漢字変換を使用しない」に設定したときの入力方法です。

4.1.1 かな漢字変換方式

かな漢字変換は、次のような特長を持っています。かな漢字変換の方法については「4.2 かな漢字変換方式の基本操作」以降で説明します。

- **逐次自動かな漢字変換が可能**

  自由文入力モードでは、プログラムが自動的に文節を判断して変換を行います。したがって、文字を変換するために特別にキーを押す必要はありません。

- **500文字までの読みを一括で変換可能**

  最大で500文字までの読みを一度に変換できます。自由文入力変換モードでは500文字までの読みを変換し直すことができますし、文章一括変換モードでは500文字までの読みを一括で変換できます。

- **柔軟性の高い編集が可能**

  変換が確定する前であれば、読みの修正、文字種の変換、文節長の変更など、自由に編集を行うことができます。

• **コード体系を自動識別して変換**

コード入力モードでは、JIS/シフトJIS/区点の3つの漢字コード体系を自動的に識別して目的の漢字や記号に変換します。(次に説明するコード入力方式とは違います。)

• **数字の表記方法の変換が可能**

アラビア数字で入力した数字を漢数字に変換したり、3桁区切りのカンマ(,)を付けることができます。

• **郵便番号変換が可能**

郵便番号変換機能によって、郵便番号から住所を呼び出すことができます。

• **辞書の編集が可能**

かな漢字変換の辞書は、基本的な単語を登録したメイン辞書と、ユーザーが登録した単語を記録したユーザー辞書に分かれています。ユーザー辞書への単語登録や、登録した単語の削除が簡単に行えます。

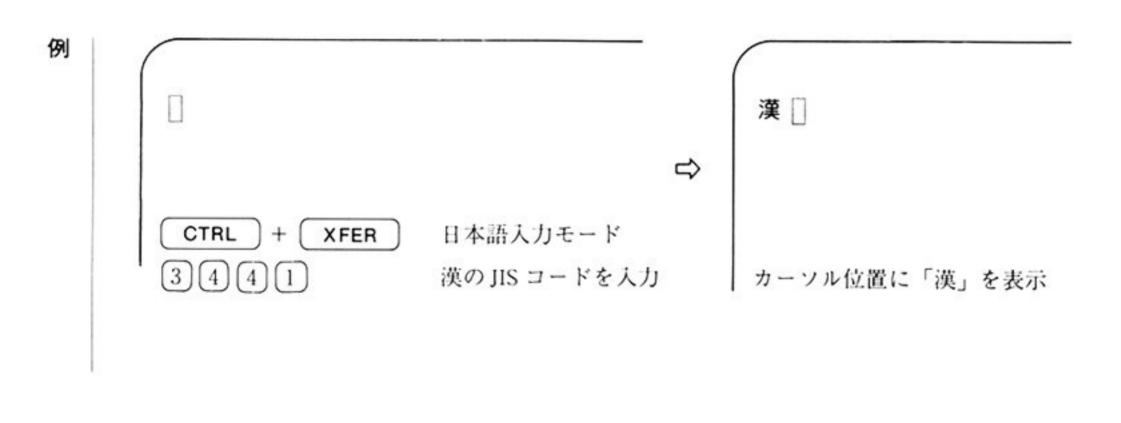
4.1.2 コード入力方式

コード入力では、1文字単位でJIS漢字コードを入力します。

① CTRL キーを押し下げたまま XFER キー(CTRL + XFER キー)を押します。
② 目的の文字のJIS漢字コードを4桁の16進数で入力します。

JIS漢字コードについては、コンピュータ本体のユーザーズマニュアルを参照してください。

かな漢字変換方式では日本語入力モードになると画面の最下行の表示が変わりますが、コード入力方式の場合は画面の表示は変わりません。また、間違ったJIS漢字コードを入力すれば、画面に何も表示されません。

例

③ CTRL + XFER キー、または ↵ キーを押すとコード入力方式を終了します。

4.2 かな漢字変換方式の基本操作

かな漢字変換方式を使って日本語入力を行うときの基本的な操作について説明します。はじめて、かな漢字変換を利用する場合に、この説明に従って操作するとかな漢字変換の基本的な操作を覚えることができます。

また、それぞれの説明の最後に「まとめ」として、その項の操作で使用した機能とそれに関係する機能をまとめました。実際に操作をしながらこれらの機能も覚えてください。

4.2.1 かな漢字変換の起動

かな漢字変換方式には2つの入力モードがあります。これらのモードは起動の方法によって選択できます。

直接入力モード

(1) 画面にカーソルが表示されている状態で CTRL + XFER キーを押します。

(2) 画面にカーソルが表示されている状態で **KINPUT** を実行します。

直接入力モードでは画面の最下行に次のような水色のガイドを表示します。

カナ　半角　英数　コード　一括　　登録　削除　ドライブ　学習　　　　学　自由　ローマ　かな　全

間接入力モード

(1) 画面にカーソルが表示されている状態で SHIFT + XFER キーを押します。

(2) 画面にカーソルが表示されていない状態で **KINPUT** を実行します。

(3) 画面にカーソルが表示されていない状態で CTRL + XFER キーを押します。

間接入力モードでは画面の最下行に次のような緑色のガイドを表示します。

カナ　半角　英数　コード　一括　　登録　削除　ドライブ　学習　　　　学　自由　ローマ　かな　全

文字を入力するとガイドは次のようになります。

編集＞　　　　　　　　　　　　　　　　　　　　学　自由　ローマ　かな　全

注意　1行の文字数を40桁にしている場合は、ファンクションキーのガイドは表示されません。画面の最下行のガイドは次のようになります。

学　自由　ローマ　かな　全

注意　間接入力モードで、画面の最下行に表示される文字数は次のとおりです。この文字数を超える文字が入力された場合、文字列の先頭が表示されなくなりますが、変換が確定したときに画面から消えていた文字もすべて画面に表示されます。

- 1行の文字数を80桁にしている場合………全角文字25文字分
- 1行の文字数を40桁にしている場合………全角文字7文字分

注意　画面にファンクションキーのガイドが表示されていない状態で、カーソルが最下行にある場合、かな漢字変換を起動すると、カーソルがかな漢字変換のガイドに隠れて見えなくなります。

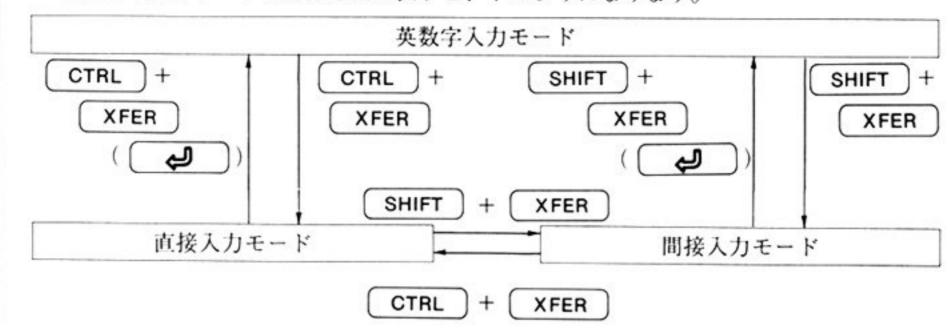
この場合、文字を入力すればガイドの上の行に入力文字が表示され、かな漢字変換も正常に行われます。

4.2.2 かな漢字変換の終了

かな漢字変換を終了するには、次の2つの方法があります。

(1) 直接入力モードのとき CTRL + XFER キー、間接入力モードのとき SHIFT + XFER キーを押します。

(2) ↵ キーを押してその行の編集を終わります。

まとめ　これらの入力モードの関係を図に表すと、下のようになります。

注意　次のような現象が起こる場合がありますが、この現象は不具合ではありません。

- 画面に表示されていた文字の一部が消えてしまう。

　未変換の文字が画面の右角までいっぱいになると、画面の下から13行分が一時的にスクロールアップします。このために一部の文字が消える場合があります。しかし、一時的に消えた文字は変換が確定すると、もとどおり表示されます。

4.2.3 基本的なかな漢字変換

かな漢字変換には、「自由文入力変換」と「文章一括変換」の２つの変換モードがあります。「自由文入力変換」では、読みの入力中であっても文節の区切りが確定したと判断された時点で、自動的にかな漢字変換が行われます。これに対して「文章一括変換」では、XFER キーを押すまで、かな漢字変換が行われません。

ここでは、次のモード設定で操作することを前提に説明をします。モード設定の違いによる操作や実行結果の違いはその都度説明します。

入力モード……直接入力モード

変換モード……自由文入力変換

操作の説明を始める前に、かな漢字変換機能の基本的な処理の流れを説明します。

① 読みを入力します。

読みは、英数字の部分は英数字で入力し、ひらがなの部分と漢字やカタカナに変換する部分はひらがなで入力します。入力した文字は緑色で表示されます。(未変換文字列)

このとき画面上で点滅しているカーソルを文字カーソルと呼びます。文字列の最後にあるときは白色、文字列の途中にあるときは緑色で表示されます。

なお、カタカナに変換する部分をはじめからカタカナで入力すると、より効率的なかな漢字変換を行うことができます。

入力する文字を切り換える方法については「入力文字の切り換え」で説明します。

② 入力した文字を変換します。

変換された文字は水色で表示されます。(変換された文字列)

変換の方法は「文字の変換」で説明します。

③ 間違って変換された文字列を、正しい候補に再変換します。

場合によっては文節の区切りも変更します。

このとき操作によって黄色で反転表示される部分があります。黄色で反転表示されている範囲がひとつの文節で、この反転部分を文節カーソルと呼びます。

変換候補の選択については「同音異義語の選択」、文節長の変更については「文節長の変更」で説明します。

④ 変換された文字列を確定します。変換が確定した文字列は白色で表示されます。
変換の確定の方法については「変換の確定」で説明します。

入力文字の切り換え

かな漢字変換では、全角のひらがなを漢字に、あるいはカタカナに変換します。したがって、ひらがな・カタカナ・英数字の混じった文章の読みを入力する際には、英数字は英数字で入力し、そのほかの部分はひらがなで入力します。また、カタカナの部分をカタカナで入力するとより効率的なかな漢字変換を行うことができます。

ここは、入力する文字の種類を切り換える方法を説明します。

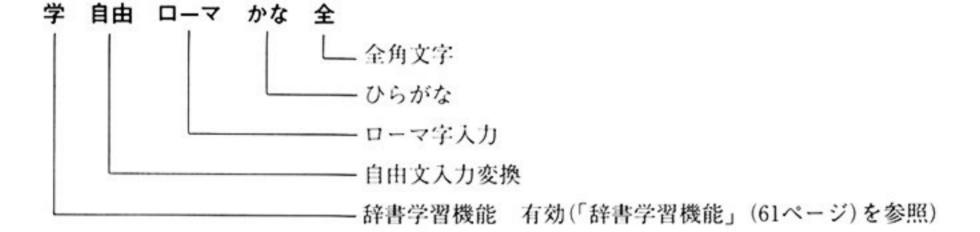
かな漢字変換を行う前に、次の2つのキーの状態を確認してください。

CAPS キー………解除状態にします。
カナ キー………解除状態にします。

かな漢字変換機能を起動します。

CTRL + XFER を押します。

かな漢字変換を初期状態のままで使用している場合、起動時にはローマ字入力で全角のひらがなを入力することができます。

入力モードと入力できる文字の種類は、次のように画面の右下に表示されます。

(1) ローマ字入力とかな入力の切り換え

ローマ字入力では、ローマ字の表記方法に従ってアルファベットでかなを入力します。かな入力ではキーボード上に印刷してあるかなを直接入力します。ローマ字入力する場合は「4.4.3　ローマ字/かな変換表」を参照してください。

ローマ字入力とかな入力の切り換えは、カナ キーによって行います。

カナ キーが解除状態………………ローマ字入力
カナ キーがロック状態……………かな入力

(2) ひらがなとカタカナの切り換え

入力文字をひらがなからカタカナに、あるいはカタカナからひらがなに切り換える方法は次のとおりです。

- CTRL + f・1 キーを押します。

また、ローマ字入力時には次の方法でひらがなとカタカナの切り換えをすることもできます。

- CAPS キーを押し、ロック状態にします。

(3) ひらがな／カタカナ入力と英数字入力の切り換え

ひらがな／カタカナ入力と英数字入力を次の方法で切り換えます。

かな入力時には [カナ] キーによってひらがな／カタカナ入力と英数字入力を切り換えます。

[カナ] キーが解除状態………………英数字入力

[カナ] キーがロック状態……………ひらがな／カタカナ入力

[カナ] キーを解除状態にしたときに、入力モードがローマ字入力の「かな」または「カナ」になった場合には「(7) ローマ字入力と英数字入力の切り換え」を参照してください。

(4) 全角と半角の切り換え

全角文字と半角文字の切り換えは、次の方法で行います。ただし、半角のひらがなは使用することはできません。ひらがな入力時に入力文字を半角にすると、自動的にカタカナの入力モードになります。

• [CTRL] + [f・2] キー、または [SHIFT] + [TAB] キーを押します

(5) 英数字の大文字と小文字の切り換え

英数字の大文字と小文字は [CAPS] キーで切り換えます。

[CAPS] キーが解除状態………………小文字入力

[CAPS] キーがロック状態……………大文字入力

また、[SHIFT] キーを押し下げたまま文字入力をすると、大文字と小文字を切り換えることができます。

(6) 英数字の全角と半角の切り換え

英数字の全角と半角の切り換えは、次の方法で行います。

• [CTRL] + [NFER] キーを押します

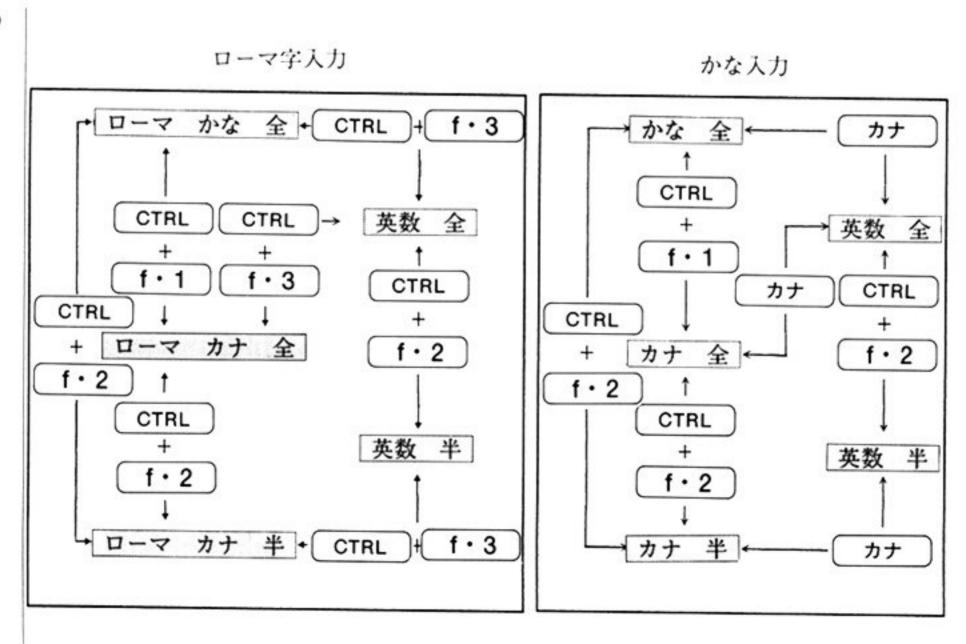
また、文字を入力中の場合は [NFER] キーだけを押しても切り換えることができます。

(7) ローマ字入力と英数字入力の切り換え

ローマ字入力と英数字入力の切り換えは、次の方法で行います。

• [CTRL] + [TAB] キー、または [CTRL] + [f・3] キーを押します

また、文字を入力中の場合は [TAB] キーだけを押しても切り換えることができます。

まとめ

読みの入力

実際に操作をしながらかな漢字変換の機能を説明していきます。

かな漢字変換は起動していますか。もし起動していない場合は、次のようにしてかな漢字変換機能を起動します。

① CTRL ＋ XFER キーを押します。

例題として「旅は道連れ、世は情。」と入力してみましょう。

まず次のように読みを入力します。次の文章はすべてひらがなですので、画面の右下には「かな」と表示されるように入力モードを設定してから入力します。ローマ字入力とかな入力は使いやすい方に設定します。(ローマ字入力の場合「4.4.3　ローマ字/かな変換表」を参照してください。)

② 「**たびはみちずれ**」と入力します。

ところが、「**道連れ**」の正しい読みは「**みちづれ**」です。いま入力した「**みちずれ**」を「**みちづれ**」に訂正します。

③ ← キーを押してカーソルを「**れ**」の位置に移します。

④ BS キーを押して「**ず**」を消します。

⑤ 「**づ**」と入力します。

⑥ → キーを押してカーソルを「**れ**」の後ろに移します。

まとめ　未変換文字列を訂正するには、次の方法があります。

- 削除したい文字の右にカーソルを移動し、BS キーを押します。
- 削除したい文字の位置にカーソルを移動し、DEL キーを押します。
- 文字を挿入したい位置にカーソルを移動し、挿入したい文字を入力します。

文字の変換

これで正しい読みが入力できましたので、続きを入力します。

⑦　「、」と入力します。

すると「**たびはみちづれ**」が「**度は道連れ**」と変換されます。かな漢字変換の自由文入力変換では、文節の区切りが確定したと判断された文節から順次かな漢字変換を行います。読点(、)をまたぐ文節はありませんので、読点が入力された時点でこの部分が変換されます。

自由文入力変換では、このほかに次の文字が入力されるとその文字の手前までの部分を変換します。

- かな記号

  スペース　、(読点)　。(句点)　「　・

- 記号

  スペース ! # $ % ([{ = + - * / : ; < > | _ ^ ~ @ ¥ , ' & ? .

もちろん、[XFER]キーを押しても、変換は行われます。

これに対して文章一括変換では、[XFER]キーを入力しない限り変換は行われません。したがって入力の途中で変換が行われることがありません。

⑧　「**よはなさけ。**」と入力します。

「、」と同じように「。」をまたぐ文節はありませんので、「**よはなさけ**」が「**世は情**」と変換されます。

まとめ

次の文字が入力されたとき、またはキーが押されたときに変換が行われます。

- 自由文入力変換時

  一部のかな記号、記号

  [XFER]キー

  自由文入力変換時には、上記の文字入力やキー操作を行わなくても、文節の区切りが確定したとかな漢字変換が判断した時点で、順次変換が行われます。

- 文章一括変換時

  [XFER]キー

同音異義語の選択

文の先頭の「**度は**」は「**旅は**」でなければなりません。そこでこれを正しく変換し直します。

⑨　[←]キーを6回押して黄色く反転したカーソルを「**度は**」の位置に移します。

⑩　[XFER]キーを押します。

画面の一番下の部分に、「**たびは**」という読みに対する変換候補の一覧が表示されます。この中から目的の変換候補を選びます。

⑪　[→]キーを押してカーソル(反転表示の部分)を「**旅は**」に合わせて[↵]キーを押します。

「**度は**」が「**旅は**」に変わります。

また、変換候補の左についている数字を入力して変換候補を選ぶこともできます。

注意 カーソルを、変換し直したい文節に移動して [↓] キーを押すと候補選択中の文節の部分につぎつぎと変換候補が表示されます。[↑] キーを押すと前の候補を表示します。このようにして目的の変換候補を選ぶこともできます。

まとめ その文節が正しい変換候補に変換されなかった場合、次の方法で正しい変換候補を選択します。

・変換候補一覧表からの選択

[XFER] キーを押します。画面の編集行に変換候補の一覧が表示されますので、カーソルキーで目的の候補にカーソルを移動する、または候補の番号を入力することによって目的の変換候補を選びます。

・文節中での選択

[↓] キーまたは [↑] キーを押します。[↓] キーを押すと次の変換候補、[↑] キーを押すと前の候補を表示しますので、目的の候補を表示させます。

変換の確定

正しく変換ができたら、その文字列を確定します。確定した文字列は再変換することはできません。

⑫ [↵] キーを押します。

変換が確定して文字列は白色に変わります。

まとめ 変換の確定を行うには、確定と同時に改行を行うかどうかによって2つの方法があります。

・確定のみを実行し、確定と同時に改行を行わない場合… [↵] キー

・確定と同時に改行を行う場合…………………………………… [SHIFT] + [↵] キー

ただし、[↵] キーの機能は「環境設定」で変更することができます。変更の方法については「第2部　BASIC ユーティリティ」の「7.1　環境設定」を参照してください。

文節長の変更

先の例題では、文節の区切り方が正しかったので文節の変換候補を選ぶだけで正しい変換ができましたが、かな漢字変換が常に文節を正しく区切るとは限りません。次の例題で、文節の区切りを間違って変換されてしまった場合の修正方法について説明します。

例題として、「ここで履物を脱いでください。」と入力してみましょう。

① [CTRL] + [XFER] キーを押してかな漢字変換を起動します。

② **「ここではきものをぬいでください。」**と入力します。

句点を入力した時点で全文が変換されます。変換結果は「**ここでは着物を脱いでください。**」となります。[←] キーを押して文節の区切りを確かめると、文節の区切りは次のようになっています。

ここでは／着物を／脱いでください／。

これではいくら [XFER] キーを押しても正しい変換候補は現われません。正しく変換を行うためには「**ここで／はきものを／ぬいでください／。**」と文節を区切らなければなりません。文節の区切りを変更するには、[SHIFT] + [←] キーまたは [SHIFT] + [→] キーを押します。

③ [HOME CLR] キーを押して文節カーソルを「**ここでは**」に移動し [SHIFT] + [←] キーを押します。

「**ここでは**」が「**ここで**」に変わり、その直後の文節も「**履物を**」に変りました。また [HOME CLR] キーは文節カーソルを、入力中または変換中の文字列の先頭に移動するときに使用します。

まとめ

文節の長さは次の操作で変更することができます。

[SHIFT] + [←] キー…文節長を1文字短くし、同時に再変換が行われます。
[SHIFT] + [→] キー…文節長を1文字長くし、同時に再変換が行われます。
[CTRL] + [←] キー…文節長を1文字短くします。このとき再変換は行われません。
[CTRL] + [→] キー…文節長を1文字長くします。このとき再変換は行われません。

読み入力中、または文節変換中に次のキーによってカーソルを移動することができます。

読み入力中

[HOME CLR] キー……………………未変換文字列の先頭に文字カーソルを移動します。
[SHIFT] + [HOME CLR] キー…未変換文字列および変換された文字列の最後の文字の後ろに文字カーソルを移動します。

文節変換中

[HOME CLR] キー……………………変換された文字列および先頭の文節に文節カーソルを移動します。
[SHIFT] + [HOME CLR] キー…未変換文字列および変換された文字列の最後の文字の後ろに文字カーソルを移動します。

変換された文字の訂正

未変換文字列に間違いがあった場合は、[BS] キーまたは [DEL] キーによって訂正することができますが、変換された文字列に間違いがあった場合は、[BS] キーや [DEL] キーだけでは訂正できません。

ここでは例題として上で修正した「ここで履物を脱いでください。」を「ここで履物を脱がないでください。」に訂正します。

④ [→] キーを押して文節カーソルを「**脱いでください**」に移動します。
⑤ [ESC] キーを押します。

「**脱いでください**」が「**ぬいでください**」とひらがなに戻り、文字の色も緑色になりました。文字の色が緑色になったということは、この部分は未変換文字列になったということです。つまりこの範囲の文字は「読みの入力」(51ページ)で説明した方法で文字を修正することができます。

⑥ [←] キーを押して文字カーソルを「**ぬ**」の右に移動します。
⑦ 「**がな**」と入力します。

「**ぬいでください**」が「**ぬがないでください**」となりました。これをもう一度変換し直します。

⑧ [XFER]キーを押します。

「ぬがないでください」が**「脱がないでください」**となりました。

⑨ [↵]キーを押して変換を確定します。

まとめ

変換された文字列を訂正するには、次の手順で訂正します。

① 次のようにして、変換された文字列の変換を取り消します。

- [ROLL UP]キーを押します。(文節カーソルのある文節の変換取り消し)
- [ESC]キーを押します。(文字カーソルの左の文節、または文節カーソルのある文節の変換取り消し)
- [ROLL DOWN]キーを押します。(文字カーソルの右の文節、または文節カーソルのある文節の変換取り消し)

② 次のようにして、変換を取り消した文節(未変換文字列)を修正します。

- 削除したい文字の右にカーソルを移動し、[BS]キーを押します。
- 削除したい文字の位置にカーソルを移動し、[DEL]キーを押します。
- 文字を挿入したい位置にカーソルを移動し、挿入したい文字を入力します。

また、次のキーを押すことによって、未変換文字列の先頭からカーソルの直前までの文字が文字種変換されます。

- [f・1]キー……ひらがなをカタカナに変換します。
- [f・2]キー……全角文字を半角文字に、半角文字に変換します。ただし、ひらがなを半角に変換すると自動的に半角のカタカナに変換されます。
- [f・3]キー……ローマ字入力で入力したひらがな/カタカナをアルファベットに変換します。ただし、次の場合は無効です。
  - かな入力の場合。
  - いったん変更された文字列を[ESC]キーや[ROLL UP]キー、[ROLL DOWN]キーなどによって未変換文字列に戻した場合。

4.2.4 記号入力

通常のかな漢字変換では入力できない記号や特殊な文字を、その記号や文字の文字コードを入力して変換します。かな漢字変換ではJIS漢字コード／シフトJISコード／区点コードのいずれのコードを入力しても、それを自動的に識別して目的の記号や漢字に変換します。ここでは、例題として郵便番号の記号「〒」をJIS漢字コード(2229)で呼び出してみましょう。JIS漢字コード、シフトJISコード、区点コードについてはコンピュータ本体のユーザーズマニュアルを参照してください。

① CTRL ＋ XFER キーを押してかな漢字変換を起動します。
② CTRL ＋ f・4 キーを押します。
③ 「**2229**」と入力します。

カーソルの位置に「〒」が表示されます。続けて文字コードを入力すれば、文字コードによる記号・文字の呼び出しが続けられますが、ここでは文字コードによる記号・文字の入力を終了します。

④ CTRL ＋ f・4 キーを押します。

文字コードによる記号・文字の入力を終了して、かな漢字変換ができる状態に戻ります。

まとめ

① 文字コードによる記号・文字の呼び出しは次の方法で開始・終了します。

- CTRL ＋ f・4 キーを押します。

使用できる文字コードは次のとおりです。

(1)JIS漢字コード

(2)シフトJISコード

(3)区点コード

ただし、区点コードを使用する場合は次の注意が必要です。

- 区の区切りには必ずカンマ(,)または読点(、)を入力します。
- 1桁の区コードおよび点コードを入力する場合でも、先頭に「0」を付けて必ず2桁で入力します。

② また、次の記号は読みによって呼び出すことができます。

読み	記号
かっこ	‘ ’ “ ” （ ） 〔 〕 ［ ］ ｛ ｝ 〈 〉 《 》 「 」 『 』 【 】
まる	○ ● ◎
しかく	◇ ◆ □ ■
ほし	☆ ★
さんかく	△ ▲ ▽ ▼
やじるし	→ ← ↑ ↓
きごう	￣ ＿ ヽ ヾ ゝ ゞ 仝 々 〆 〇 ー ― ‐ ／ ＼ ～ ‖ ｜ ＋ － ± × ÷ ＝
	≠ ＜ ＞ ≦ ≧ ∞ ∴ ♂ ♀ ° ′ ″ ℃ ￥ ＄ ¢ £ ％ ＃ ＆ ＊ ＠ ※ 〒
てん	、 。 ， ． ・ ： ； ？ ゛ ゜ ´ ｀ ¨ ＾ … ‥
ぎりしゃ	Α Β Γ Δ Ε Ζ Η Θ Ι Κ Λ Μ Ν Ξ Ο Π Ρ Σ Τ Υ Φ Χ Ψ Ω
	α β γ δ ε ζ η θ ι κ λ μ ν ξ ο π ρ σ τ υ φ χ ψ ω
ろしあ	А Б В Г Д Е Ё Ж З И Й К Л М Н О П Р С Т У Ф Х Ц
	Ч Ш Щ Ъ Ы Ь Э Ю Я
	а б в г д е ё ж з и й к л м н о п р с т у ф х ц
	ч ш щ ъ ы ь э ю я

4.2.5 郵便番号変換

かな漢字変換方式では、郵便番号からその郵便番号に該当する住所を呼び出すことができます。これを郵便番号変換と呼び、どの入力モードからでも GRPH + TAB キーで実行することができます。

① GRPH + TAB キーを押します。
② 「**399**」と入力します。
③ XFER キーを押します。

いま入力した郵便番号「**399**」に該当する住所が表示されます。この場合は「**長野県松本市**」と「**長野県塩尻市**」の2つの候補が表示されます。これは ← → キーで選ぶことができますし、1つの画面に候補がすべて表示できない場合は ↓ キーで次のページの候補を表示することもできます。ここでは「**長野県松本市**」を選びます。

④ そのまま ↵ キーを押します。

カーソル位置に「**長野県松本市**」と表示され、もとの入力モードに戻ります。

まとめ　郵便番号変換は次の方法で実行します。
- GRPH + TAB キーを押します。
候補の住所に県名を入れるか省略するかは、郵便番号入力後、XFER キーを押す前に ↑ キーで選択します。

注意　1行の文字数を40桁にしている場合、画面の一番下の行には「＝」しか表示されません。

BMENUの「日本語入力機能メンテナンス」の「環境設定」の中で「かな漢字変換と郵便番号変換を使用する」に設定しないと郵便番号変換を行うことはできません。

4.2.6 単語登録

かな漢字変換機能の辞書には約55,000語の単語が登録されていますが、この辞書に登録されていない専門用語などの単語は、ユーザー辞書として登録し、登録した読みを使って通常のかな漢字変換で呼び出すことができます。また、ユーザー辞書には記号なども登録できますので、頻繁に使用する記号を単語登録しておけば、コード入力を使用しないで記号を入力することができます。

ここでは郵便番号記号「〒」を「ゆうびん」という読みで登録してみましょう。「4.2.4　記号入力」を参照して「〒」を呼び出してから、次のようにしてください。

① CTRL + f・6 キーを押します。
② ← キーを押してカーソルを「〒」の位置に移動します。
③ SHIFT + → キーを押して「〒」を反転表示させます。

いま反転表示されている部分が辞書に登録されます。

④ ↵ キーを押します。
⑤ 読みを入力します。ここでは「**ゆうびん**」と入力して ↵ キーを押します。

次に登録する単語の品詞を選びます。ここで選べる品詞の種類は次のとおりです。

1．普通…………通常の名詞などで指定します。通常は「1．普通」にします。
2．地名…………地名として扱われます。
3．サ変…………サ変動詞。つまり、名詞の下に「する」がついて動詞として機能する名詞です。
4．人名(姓)……人名の姓です。
5．人名(名)……人名の名です。

これらの品詞は、ひとつの単語について複数指定することもできます。

ここでは、「1．普通」で登録します。

注意 ある単語をサ変動詞として登録する場合、品詞に「サ変」と「普通」を選ぶとサ変動詞以外としても呼び出すことができます。例えば「リストア」という単語を「りすとあ」という読みで登録して、「普通」と「サ変」で登録すると、「りすとあする」を「リストアする」と変換するほか、「りすとあは」を「リストアは」と変換することもできます。

⑥ 「1」と入力します。
⑦ [↵] キーを押します。

以上でユーザー辞書登録の操作は終わりです。

まとめ ユーザー辞書の登録は次の手順で行います。

① 登録する単語を表示した状態で [CTRL] + [f・6] キーを押します。
② カーソルキーを使って登録する単語の先頭にカーソルを移動します。
③ [SHIFT] を押し下げたまま、[→] キーまたは [←] キーを押して登録する単語の範囲を指定します。
④ [↵] キーを押します。
⑤ 登録する単語の品詞の番号を入力します。選択できる品詞は次のとおりです。

1．普通　2．地名　3．サ変　4．人名(姓)　5．人名(名)

ユーザー辞書登録時の制限は次のとおりです。

- 登録可能単語数……………………約1,000語～4,000語(環境設定で変更可能)
- 1単語当たりの文字数……………8文字以内
- 1単語当たりの読みの文字数……12文字以内
- 半角文字を1文字でも含む単語の登録はできません。

注意 1行の文字数を40桁にした場合、[CTRL] + [f・6] キーを押しても単語の登録は行えません。

4.2.7 数字の変換

かな漢字変換では、アラビア数字で入力した数字を [XFER] キーを使って漢数字に変換したり、三桁区切りのカンマ(,)を入れることができます。

ここでは、この機能を使って「十二万三千四百五十六」と入力してみましょう。

① CTRL ＋ XFER キーを押して、かな漢字変換機能を起動します。

② 「**123456**」と入力します。

③ XFER キーを押します。

④ ← キーを押して文節カーソルを「**123456**」に合わせます。

⑤ XFER キーを押します。

「**123456**」が「**十二万三千四百五十六**」に変換されます。

まとめ かな漢字変換機能ではアラビア数字を次のように変換します。

・アラビア数字	**例** 123456
→漢数字(単位あり)	→十二万三千四百五十六
→漢数字(旧字体：単位あり)	→拾弐万参千四百五拾六
→漢数字(単位なし)	→一二三四五六
→全角数字の三桁区切りのカンマ(全角)入り	→１２３，４５６
→全角数字の三桁区切りのカンマ(半角)入り	→１２３,４５６
→半角数字の三桁区切りのカンマ(半角)入り	→123,456
→アラビア数字に戻る	→123456(戻り)

(ただし、３桁以下の数の場合は三桁区切りのカンマ(全角／半角共に)入りは省略されます。)

注意 「123.456」のように小数点(ピリオド)を含む数字を入力した場合、ピリオドは文節の区切りとして判断されるため、「123」「.」「456」の３つの文節に分かれてしまいます。
したがって、このような数字の場合は、「123」(整数部)と「456」(小数部)を別々に変換しなければなりません。

4.3　かな漢字変換の応用操作

ここでは、かな漢字変換をより高度に使いこなすための機能を説明します。これらの機能を覚えることによって、より効率よくかな漢字変換を使用することができます。基本操作を覚えたら、次にここで説明する機能も使ってみましょう。

4.3.1　操作環境の変更

かな漢字変換では、かな漢字変換使用中に変換モード・辞書ドライブ/辞書名・辞書学習機能といった操作環境を変更することができます。ここでは、これらの操作環境を変更する方法について説明します。

変換モードの変更

変換モードには自由文入力変換と文章一括変換があります。変換モードは次のようにして切り換えます。

① CTRL + f・5 キーを押します。

現在の変換モードは画面の右下に表示されます。

辞書ドライブ/辞書名の変更

かな漢字変換用の辞書が登録されているドライブと辞書名は、環境設定によって起動時に決まっています。しかし、この辞書が登録されているドライブと辞書名を次の方法で変更することができます。

① CTRL + f・8 キーを押します。
② 画面の最下行に現在の辞書ドライブと辞書名が表示されます。
③ BS キーを押して辞書名と辞書ドライブを消して、新しく辞書ドライブと辞書名を入力します。
入力の書式は次のとおりです。なお、辞書名は拡張子を入力する必要はありません。
　ドライブ番号：辞書名
また、次のように入力することもできます。
　ドライブ番号：……辞書名は変えずに辞書ドライブだけ変更します。
　辞書名………………辞書ドライブは変えずに辞書名だけ変更します。
④ ↵ キーを押します。

ドライブ番号/辞書名を入力しないで ↵ キーを押すと、辞書ドライブは変更されません。

辞書学習機能の変更

辞書学習機能というのは、かな漢字変換で選択された単語を辞書に反映させ、次の変換時には前回選択された単語を優先的に変換候補にする機能です。この辞書学習機能を有効にするか、無効にするかは次のようにして切り替えます。

① CTRL ＋ f・9 キーを押します。

画面の右下に「**学**」と表示されていれば辞書学習機能は有効です。この表示がない場合は辞書学習機能は働いていません。辞書学習機能を有効にした場合は、辞書の入っているディスクを書き込み可能な設定にしていください。

参考　かな漢字変換機能では、変換候補の優先順位を決める際に、ここで説明した辞書の学習結果とともに、変換候補の品詞を参考にしています。したがって、前回選択した候補が必ず最初の変換候補になるとは限りません。

4.3.2 ユーザー登録単語の削除

ユーザーが登録した単語が必要なくなった場合、これを削除することもできます。削除は次の方法で行います。

① CTRL ＋ f・7 キーを押します。
② 削除したい単語の読みを入力して XFER キーを押します。
③ 入力した読みで登録されたユーザー登録単語の一覧が表示されますので、削除したい単語の番号を入力します。または ← → キーで選択して ↵ キーを押します。一度にすべての単語を表示できない場合は ↓ ↑ キーで前ページ、次ページの単語を表示することができます。
④ 確認のメッセージを表示しますので、削除してよい場合は「**Y**」、削除を中止する場合は「**N**」と入力します。

注意　削除したユーザー登録単語は、再登録する以外にもとに戻す方法がありません。ユーザー登録単語の削除は十分に注意して行ってください。
1行の文字数を40桁にしている場合、CTRL ＋ f・7 を押しても単語の削除は行えません。

4.4　かな漢字変換方式に関する資料

4.4.1　キーの機能一覧

かな漢字変換機能では、キーボードからの入力でモードを切り換えたり、さまざまな変換操作などを行います。ここでは、次の4つの機能に大別してキーに割り当てられた機能を説明します。

■モードの切り換え（起動／終了）

かな漢字変換機能の入力モードなど、動作の状態を切り換えます。

■カーソルの移動

文字を入力する位置、変換する文字列などを示すカーソルを移動します。

■文字列の変換

文字列を目的の単語に変換します。

■ユーザー辞書の編集

辞書に登録されていない単語を登録したり、不要になった単語を削除します。

また、これらの機能は、操作のどの段階にあるかによって有効なときと、有効でないときがあります。ここでは、次の５つの状態に分けて、その機能が有効かどうかを示します。

①未入力時

変換のための文字列が入力されていない状態です。カラーディスプレイを使用している場合、水色や緑色の文字、あるいは黄色の反転文字は表示されていません。

②読み入力時

変換のための読みを入力している状態です。カラーディスプレイを使用している場合、白色の文字カーソルが表示されています。

③未変換文字列編集時

未変換文字列を修正、あるいは文字種変換している状態です。ディスプレイを使用している場合、文字カーソルは緑色、または水色で表示されています。

④文節編集時

文節に対して操作をしている状態です。カラーディスプレイを使用している場合、黄色で反転した文節カーソルが表示されています。

⑤同音異義語選択時

直接入力モードで、同音異義語から変換候補を選択している状態です。画面の編集行には、変換候補の一覧が表示されています。

例

	キー	①	②	③	④	⑤
直接入力モードの起動／終了	CTRL + XFER	○	×	×	×	×

①未入力時に有効
②読み入力時に無効
③未変換文字列編集時に無効
④文節編集時に無効
⑤同音異義語選択時に無効

モードの切り替え(起動／終了)

	キー	①	②	③	④	⑤
直接入力モードの起動／終了	CTRL + XFER	○	×	×	×	×
(変換中の文字列を確定)	CTRL + XFER	×	○	○	○	○
間接入力モードの起動／終了	SHIFT + XFER	○	×	×	×	×
(変換中の文字列を確定)	SHIFT + XFER	×	○	○	○	×
郵便番号変換モードの起動	GRPH + TAB	○	×	×	×	×
コード入力モードの起動	f・4	×	○	○	○	×
	CTRL + f・4	○	○	○	○	×
ローマ字入力／かな入力の切り換え	カナ	○	○	○	○	×
ひらがな入力／カタカナ入力の切り換え	CTRL + f・1	○	○	○	○	×
(ローマ字入力時)	CAPS	○	○	○	○	×
(ローマ字入力時)	SHIFT + 文字キー	○	○	○	○	×
全角入力／半角入力の切り換え	CTRL + f・2	○	○	○	○	×
	SHIFT + TAB	○	○	○	○	×
ローマ字入力／英数字入力の切り換え(ローマ字入力時)	CTRL + f・3	○	○	○	○	×
	TAB	×	○	○	○	×
	CTRL + TAB	○	○	○	○	×
英数字の全角/半角切り換え	CTRL + NFER	×	○	○	○	×
	NFER	○	○	○	○	×
自由文入力変換／文章一括変換の切り換え	f・5	×	○	○	○	×
	CTRL + f・5	○	○	○	○	×
辞書ドライブの切り換え	f・8	×	○	○	○	×
	CTRL + f・8	○	○	○	○	×
辞書学習機能の有効／無効の切り換え	f・9	×	○	○	○	×
	CTRL + f・9	○	○	○	○	×
大文字入力／小文字入力の切り換え(英数字入力時)	CAPS	○	○	○	○	×
	SHIFT + 文字キー	○	○	○	○	×

カーソルの移動

文字カーソルの移動

操作	キー	①	②	③	④	⑤
1文字左に移動	←	×	○	△	×	×
	SHIFT + ←	×	○	△	×	×
	CTRL + ←	×	○	△	×	×
(△：カーソルが未変換文字列の途中にあるとき)						
1文字右に移動	→	×	○	△	×	×
	SHIFT + →	×	○	△	×	×
	CTRL + →	×	○	△	×	×
(△：カーソルが未変換文字列の途中にあるとき)						
編集中の文節の先頭に移動(文節カーソルの左が未変換文字列のとき)	←	×	×	×	○	×
右にある未変換文字列の先頭に移動(文節カーソルの右が未変換文字列のとき)	→	×	×	×	○	×

文節カーソルの移動

操作	キー	①	②	③	④	⑤
左の文節に移動						
(カーソルが未変換文字列の先頭にあるとき)	←	×	×	○	×	×
	SHIFT + ←	×	×	○	×	×
	CTRL + ←	×	×	○	×	×
(文節カーソルの左が変換された文字列のとき)	←	×	×	×	○	×
右の文節に移動						
(カーソルが未変換文字列の末尾にあるとき)	→	×	×	○	×	×
	SHIFT + →	×	×	○	×	×
	CTRL + →	×	×	○	×	×
(文節カーソルの右が変換された文字列のとき)	→	×	×	×	○	×

先頭位置へのカーソル移動

操作	キー	①	②	③	④	⑤
文字列の先頭に移動	HOME CLR	×	○	△	○	×
(△：カーソルが未変換文字列の先頭にあるとき)						
	↓	×	○	△	×	×
(△：カーソルが未変換文字列の末尾にあり、変換された文字列がないとき)						
編集中の未変換文字列の先頭に移動	↑	×	○	○	×	×
(カーソルが未変換文字列の途中にあるとき)	HOME CLR	×	×	○	×	×
編集中の文節の先頭に移動(文節カーソルの左が未変換文字列のとき)	←	×	×	×	○	×
右にある未変換文字列の先頭に移動(文節カーソルの右が未変換文字列のとき)	→	×	×	×	○	×

末尾位置へのカーソル移動

操作	キー	①	②	③	④	⑤
文字列の末尾に移動	SHIFT + HOME CLR	×	○	△	○	×
(カーソルが未変換文字列の末尾にあるとき)						
(カーソルが未変換文字列の途中にあるとき)	↓	×	×	○	×	×
編集中の未変換文字列の末尾に移動(カーソルが未変換文字列の途中にあるとき)	SHIFT + HOME CLR	×	×	○	×	×

文字の削除

操作	キー	①	②	③	④	⑤
カーソルの左の1文字を削除	BS	×	○	○	×	×
カーソル位置の1文字を削除	DEL	×	○	○	×	×
入力した文字列をすべて削除	SHIFT + ESC	×	○	○	×	×

文字列の変換

かな漢字変換の起動

操作	キー	①	②	③	④	⑤
かな漢字変換の起動	XFER / スペース	×	○	○	×	×

変換候補の選択

操作	キー	①	②	③	④	⑤
次候補を表示	↓	×	△	▲	○	×
(△：変換された文字列があるとき)						
(▲：カーソルが未変換文字列の末尾にあり、変換された文字列があるとき)						
(間接入力モード時)	XFER	×	×	×	○	×
前候補を表示	↑	×	×	×	○	×
同音異義語の選択を開始(直接入力モード時)	XFER	×	×	×	○	×
同音異義語の中から左の候補を選択	SHIFT + XFER	×	×	×	×	○
	←	×	×	×	×	○
	SHIFT + ←	×	×	×	×	○
	CTRL + ←	×	×	×	×	○
	BS	×	×	×	×	○
同音異義語の中から右の候補を選択	XFER	×	×	×	×	○
	→	×	×	×	×	○
	SHIFT + →	×	×	×	×	○
	CTRL + →	×	×	×	×	○
変換候補一覧の前ページを表示	↑	×	×	×	×	○
変換候補一覧の次ページを表示	↓	×	×	×	×	○
同音異義語の中から変換候補を決定	↵	×	×	×	×	○
同音異義語の選択を中止	ESC	×	×	×	×	○
	SHIFT + ESC	×	×	×	×	○

文字種の変換

操作	キー	①	②	③	④	⑤
ひらがな↔カタカナの変換	f・1	×	○	○	×	×
全角↔半角の変換	f・2	×	○	○	×	×
ひらがな／カタカナ↔アルファベットの変換	f・3	×	○	○	×	×

	キー			①	②	③	④	⑤
文節長の変更								
文節長を短くする(再変換を行う)……	SHIFT	+	←	×	×	×	○	×
(再変換を行わない)……	CTRL	+	←	×	×	×	○	×
文節長を長くする(再変換を行う)……	SHIFT	+	→	×	×	×	○	×
(再変換を行わない)……	CTRL	+	→	×	×	×	○	×
変換の取り消し								
カーソルのある文節の変換を取り消す……	ESC			×	×	×	○	×
	SHIFT	+	ESC	×	×	×	○	×
	ROLL UP			×	×	×	○	×
	ROLL DOWN			×	×	×	○	×
カーソルの左の文節の変換を取り消す……	ROLL UP			×	○	○	×	×
(カーソルの左が変換された文字列のとき)……	BS			×	○	○	×	×
カーソルの右の文節の変換を取り消す……	ROLL DOWN			×	○	○	×	×
文字列の確定								
文字列の確定……	SHIFT	+	↵	×	○	○	○	×
	↵			×	○	○	○	○

ユーザー辞書の編集	キー			①	②	③	④	⑤
単語の登録……	f・6			×	○	○	○	×
	CTRL	+	f・6	○	○	○	○	×
ユーザー登録単語の削除……	f・7			×	○	○	○	×
	CTRL	+	f・7	○	○	○	○	×

4.4.2 エラーメッセージ一覧

かな漢字変換操作中に、次のようなエラーメッセージを表示することがあります。このような場合には、次のような方法で対処してください。

辞書ファイルがありません

原因：指定したドライブに辞書ファイルがありません。

対策：①指定したドライブに辞書が記録されているディスクを挿入し、辞書ドライブを変更してください。

②辞書があるドライブを辞書ドライブに変更してください。

③辞書名を正しい名称に変更してください。

登録範囲を指定してください

原因：範囲を指定せずに単語を辞書登録しようとしています。

対策：範囲を指定してから登録してください。

ディスケットが書込み禁止

原因：フロッピーディスクがライトプロテクト状態になっています。

対策：①フロッピーディスクのライトプロテクトを解除してください。

②辞書学習機能を「しない」にしてください。

ユーザー辞書に登録されていません

原因：辞書削除の際に単語の読みの入力を間違えた、または登録されていない単語を削除しようとしています。

対策：正しい読みを入力してください。

最初に使用した辞書と違います

原因：かな漢字変換を起動した直後に使用した辞書とは違う辞書に変更しようとしています。

対策：指定した辞書ドライブにかな漢字変換機能の立ち上げに使用した辞書を入れて、辞書ドライブを変更し、［↵］キーを押します。

ユーザー辞書の容量がいっぱいです

原因：ユーザー辞書容量がオーバーしています。

対策：日本語辞書メンテナンスユーティリティを使って、ユーザー辞書を再編成してください。

ユーザー辞書がありません

原因：郵便番号ユーザー辞書の表示プログラムを直接実行するとき、指定したドライブにユーザー辞書がありません。

対策：郵便番号ユーザー辞書が登録されているドライブを正しく指定して再実行してください。

4.4.3 ローマ字／かな変換表

あ行	あ	い	う	え	お
	A	I	U	E	O
か行	か	き	く	け	こ
	KA CA	KI	KU CU	KE	KO CO
さ行	さ	し	す	せ	そ
	SA	SI SHI CI	SU	SE CE	SO
た行	た	ち	つ	て	と
	TA	TI CHI	TU TSU	TE	TO
な行	な	に	ぬ	ね	の
	NA	NI	NU	NE	NO
は行	は	ひ	ふ	へ	ほ
	HA	HI	HU FU	HE	HO
ま行	ま	み	む	め	も
	MA	MI	MU	ME	MO
や行	や		ゆ		よ
	YA		YU		YO
ら行	ら	り	る	れ	ろ
	RA LA	RI LI	RU LU	RE LE	RO LO
わ行	わ	ゐ	う	ゑ	を
	WA	WI	WU	WE	WO
ん	ん				
	N　NN　XN　M　MN				
が行	が	ぎ	ぐ	げ	ご
	GA	GI	GU	GE	GO
ざ行	ざ	じ	ず	ぜ	ぞ
	ZA	ZI JI	ZU	ZE	ZO
だ行	だ	ぢ	づ	で	ど
	DA	DI	DU	DE	DO
ば行	ば	び	ぶ	べ	ぼ
	BA	BI	BU	BE	BO
ぱ行	ぱ	ぴ	ぷ	ぺ	ぽ
	PA	PI	PU	PE	PO
いぇ				いぇ	
				YE	
うぁ行	うぁ	うぃ	うぅ	うぇ	うぉ
	WHA	WHI	WHU	WHE	WHO
うぁ行	うぁ	うぃ	う	うぇ	うぉ
	VA	VI	VU	VE	VO
きゃ行	きゃ	きぃ	きゅ	きぇ	きょ
	KYA	KYI	KYU	KYE	KYO

ぎゃ行	ぎゃ GYA	ぎぃ GYI	ぎゅ GYU	ぎぇ GYE	ぎょ GYO
くぁ行	くぁ QA	くぃ QI	くぅ QU	くぇ QE	くぉ QO
ぐぁ行	ぐぁ GWA	ぐぃ GWI	ぐぅ GWU	ぐぇ GWE	ぐぉ GWO
しゃ行	しゃ SYA SHA	しぃ SYI	しゅ SYU SHU	しぇ SYE SHE	しょ SYO SHO
じゃ行	じゃ ZYA JYA JA	じぃ ZYI JYI	じゅ ZYU JYU JU	じぇ ZYE JYE JE	じょ ZYO JYO JO
ちゃ行	ちゃ CYA CHA TYA	ちぃ CYI TYI	ちゅ CYU CHU TYU	ちぇ CYE CHE TYE	ちょ CYO CHO TYO
ぢゃ行	ぢゃ DYA	ぢぃ DYI	ぢゅ DYU	ぢぇ DYE	ぢょ DYO
でゃ行	でゃ DHA	でぃ DHI	でゅ DHU	でぇ DHE	でょ DHO
つぁ行	つぁ TSA	つぃ TSI	つぅ	つぇ TSE	つぉ TSO
てゃ行	てゃ THA	てぃ THI	てゅ THU	てぇ THE	てょ THO
にゃ行	にゃ NYA	にぃ NYI	にゅ NYU	にぇ NYE	にょ NYO
ひゃ行	ひゃ HYA	ひぃ HYI	ひゅ HYU	ひぇ HYE	ひょ HYO
びゃ行	びゃ BYA	びぃ BYI	びゅ BYU	びぇ BYE	びょ BYO
ふぁ行	ふぁ FA	ふぃ FI		ふぇ FE	ふぉ FO
ふゃ行	ふゃ FYA	ふぃ FYI	ふゅ FYU	ふぇ FYE	ふょ FYO
ぴゃ行	ぴゃ PYA	ぴぃ PYI	ぴゅ PYU	ぴぇ PYE	ぴょ PYO
みゃ行	みゃ MYA	みぃ MYI	みゅ MYU	みぇ MYE	みょ MYO
りゃ行	りゃ RYA LYA	りぃ RYI LYI	りゅ RYU LYU	りぇ RYU LYU	りょ RYO LYO
ぁ行	ぁ XA	ぃ XI XYI	ぅ XU XWU	ぇ XE XYE	ぉ XO XWO
ヵ行	ヵ XKA			ヶ XKE	

っ			っ XTU		
ゃ行	ゃ XYA		ゅ XYU		ょ XYO
ゎ行	ゎ XWA	ぃ XWI	ぅ XWU	ぇ XWE	ぉ XWO

注意
(1) "XKA"、"XKE"はカタカナのときのみ「ヵ」、「ヶ」となり、ひらがなの場合は「か」、「け」、半角では「ｶ」「ｹ」となります。
(2) "XWA"は全角のときのみ「ゎ」、「ヮ」となり、半角では「ﾜ」となります。

4.4.4 使用するファイル

かな漢字変換の動作に必要なファイルを示します。

日本語入力機能はBMENUの「フォーマット」の中で「システムディスク作成」を実行したときにディスクの特別な場所に書き込まれます。この機能は「日本語機能メンテナンス」の「環境設定」の中で、日本語入力機能を「使用する」に設定すると、BASIC起動時に読み込まれて、かな漢字変換を利用することができます。この他に辞書として次のファイルが必要になります。

egbdic.man	かな漢字変換のためのメイン辞書です。辞書名は変更可能。ただし拡張子は.manでなければなりません。また、拡張子を除くファイル名がユーザー辞書と同じでなければなりません。
egbdic.usr	かな漢字変換のためのユーザー辞書です。ユーザーが登録した単語が格納されています。辞書名は変更可能。ただし拡張子は.usrでなければなりません。また、拡張子を除くファイル名がメイン辞書と同じでなければなりません。
egzipm.dic	郵便番号変換のためのメイン辞書です。
egzipu.dic	郵便番号変換のためのユーザー辞書です。ユーザーが登録した住所が格納されています。

4.4.5 仕様

かな漢字変換	自由文入力変換、文章一括変換
漢数字変換	5種類
入力モード	ローマ字ひらがな／カタカナ、ローマ字半角カタカナ、ひらがな／カタカナ、英数全角／半角、半角カタカナ
入力文字の種類	すべての文字種を混在入力可能
文字種変換	英数変換(ローマ字入力時のみ)、ひらがな変換、カタカナ変換、全角変換、半角変換
コード入力	JIS、シフトJIS、区点コードの自動判別
一度に入力できる文字数	文章一括変換時：全角で約500文字(1行の文字数＝80桁) 文章一括変換時：全角で250文字(1行の文字数＝40桁) 自由文入変換時：読みで約500文字(全角)
一度に変換できる最大文字数	約500文字(全角ひらがな)
変換できる文字種	全角ひらがな、文字種混在での変換可能

郵便番号変換	市、区、町、村、郡をサポート
再変換	500文字(全角)を確定するまでは全文または任意の部分で再変換が可能
編集	500文字(全角)の文章の全文、または任意の部分で挿入、削除が可能
文法解析	二文節最長一致法、接頭語処理、接尾語処理、活用語処理、数詞処理、助数詞処理、副詞処理、連体詞処理、人名/地名処理
文節の修正	一文字単位の修正、任意の区切りでの修正
文節長学習	可能(文節長の変更を登録)
辞書の語数	約55,000語(約420KB)
郵便番号の件数	約7,000件、56KB (財団法人ホスピタルサービスセンター発行郵便番号簿を収録)
ユーザー登録語数	約1,000語～4,000語
ユーザー単語登録の方法	画面上の任意の文字が登録可能
ユーザー登録単語の文字数	1単語全角8文字
ユーザー登録単語の読みの文字数	1単語全角12文字
ユーザー登録単語の読み文字種	全角ひらがな、および全角ひらがなに続く英数字/記号
ユーザー登録単語の削除	可能
ユーザー登録郵便番号辞書の件数	約500件
学習機能	単語学習/文節長学習
辞書ドライブ/辞書名変更	可能(ハードディスク対応可能)ただし、辞書のファイル拡張子はman/usrの組み合わせのみ。

ユーティリティソフトBMENUの日本語入力機能メンテナンス

「環境設定」	かな漢字変換の使用の有無 かな漢字変換起動時の動作環境の設定
「辞書ユーティリティ」	ユーザー辞書とメイン辞書の結合 ユーザー辞書の再編成 ユーザー辞書の一覧出力
「郵便番号辞書」	郵便番号ユーザー辞書への登録、削除
「郵便番号ユーザー辞書表示」	郵便番号ユーザー辞書の一覧表示

第5章

入出力装置とファイル

BASICでは、さまざまな入力装置をすべて「ファイル」という共通の考え方で扱っています。共通の考え方をすることにより、異なるタイプの入出力装置であっても「ファイル」を操作するプログラムを作成すればよく、プログラムの作成効率が向上します。

5.1 ファイル

ファイルとはプログラムやデータなどの情報の集まりです。BASICではディスク、通信回線などの周辺機器と情報のやり取りを行う際に、このファイルの形式で作業を行います。このため周辺機器とデータの入出力を行う際には、どの装置のどのファイルとやり取りを行うのかを指定する必要があります。この指定をするものがファイル指定子です。ファイル指定子は次の形式を持ちます。

ファイル指定子＝"［デバイス名:］［ファイル名］"

デバイス名は周辺装置を指定するもので、省略すると1（ディスクドライブ1）を指定したことになります。ファイル名はデバイス上のファイルを指定するものです。

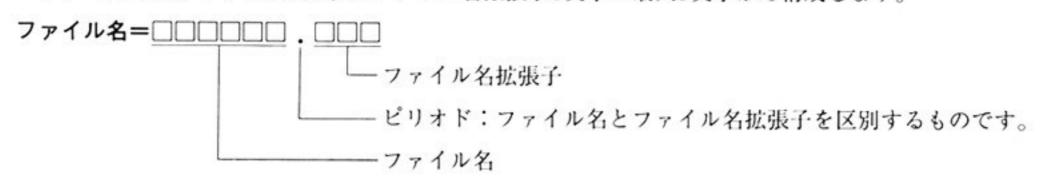
5.1.1 デバイス名

BASICで使用する周辺装置には、それぞれ次のデバイス名を付けています。周辺機器の内容によって入力専用、出力専用、入出力可能なものがあります。

デバイス名	周辺機器名	入力	出力
KYBD	キーボード	○	×
SCRN	画面	×	○
LPT1 または LPT	プリンタ	×	○
1 または省略	ディスク 1	○	○
2	ディスク 2	○	○
3	ディスク 3	○	○
4	ディスク 4	○	○
5	ディスク 5	○	○
6	ディスク 6	○	○
7	ディスク 7	○	○
8	ディスク 8	○	○
9	ディスク 9	○	○
10	ディスク10	○	○
COM1 または COM	通信回線 1 (内蔵)	○	○
COM2：	通信回線 2	○	○
COM3：	通信回線 3	○	○

○：使用可能
×：使用不可

5.1.2 ファイル名

ファイル名はファイル名6文字とファイル名拡張子3文字の最大9文字から構成します。

ファイル名は、2つ以上のファイルを持つことのできるデバイスに対して、そのデバイス上のどのファイルに対して入出力を行うのかを決めるものです。ファイル名は最大6文字に、ファイル名拡張子3文字を加えた9文字で区別されます。ファイル名に6文字以上を指定した場合は7文字目以降はファイル名拡張子として扱われます。ファイル名に使用できるのは英数字とカタカナです。漢字のような2バイト文字を使用することはできません。また英字については大文字と小文字の区別を行いますので注意してください。

ファイル名で使用可能な文字

0，1，・・・，9

A，B，・・・，Z

a，b，・・・，z

ア，イ，・・・，ン

例
```
LOAD "TEST.BAS"
LOAD "test.bas"
```
TEST. BAS と test. bas は別のファイルとして区別されます。

ファイル名拡張子は、一般にファイル内容を区別するために用います。例えば、プログラムファイルであればBASを、データファイルであればDATをなどと自分でわかりやすい拡張子を決めておくと、あとになってどのファイルが何のためのファイルかが一目でわかります。ファイル名拡張子は省略しても構いません。

5.2 ファイルのオープンとクローズ

ファイル指定子は「どこの装置の、何という名前のファイルか」を指定します。さらにこのファイルに対してデータの入出力を行うにはどうすればよいのかを説明します。

5.2.1 OPEN

新しいファイルを作成したり、すでに作成してあるファイルからデータを読み出したり追加したりするには、まず、ファイルを開く(オープンする)ことが必要です。これはノートに何かを書き込むために、まず、そのノートを開くという動作と同じです。

ファイルを開くためのBASICの命令は、OPEN文です。OPEN文ではファイル指定子とそのファイルをどういうモードで使用するのかを指定します。このモードはファイルに対してどのような処理を行うかを宣言するもので次の4つがあります。

モード	処理	ファイルの種類
INPUT	既存のファイルからデータを順番に読み出します。	シーケンシャルファイル
OUTPUT	新しいファイルを作成して、データを順番に書き込みます。	
APPEND	既存のファイルの後ろにデータを追加します。	
省略	レコード番号によるデータの読み出しと書き込みを行います。	ランダムファイル

指定したファイルが存在しない場合、次のよう処理を行います。

INPUT	File not found（ファイルが存在しない）エラーが発生します。
OUTPUT	常に新しいファイルを作成します。したがって、すでに指定した名前のファイルが存在した場合は、そのファイルの内容はすべて失われ、ファイルの先頭から改めてデータが書き込まれていきます。
APPEND	File not found（ファイルが存在しない）エラーが発生します。
省略	新しくファイルを作ります。

5.2.2　CLOSE

オープンしたファイルに対する処理が終了したら、必ずファイルを閉じる必要があります。これも使ったノートを閉じることと同様です。

ファイルを閉じるための命令はCLOSE 文です。

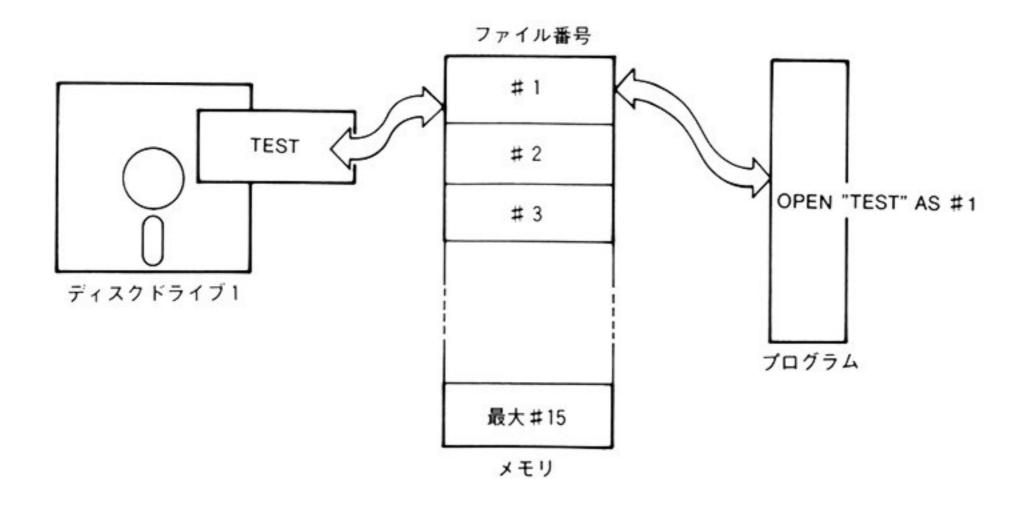
注意　ファイルをオープンした状態で、ディスクの交換を行わないでください。ディスクの交換を行う場合は、必ずファイルをクローズしてください。

5.2.3　ファイル番号

ファイルの入出力はファイル番号を用いて行います。ファイル番号は入出力を行うためにファイルをオープンした際に、ファイルに対して割り当てる固有の番号です。

ファイルの入出力を行うときには、そのファイル専用の窓口（バッファ）を設定します。つまり、ファイルにデータを書き込むとき、あるいは読み出すときはそのファイルと一対一に対応する専用の窓口を必ず通ります。この専用の窓口の番号がファイル番号です。

ファイル番号はファイルと一対一に対応しているため、そのファイルをクローズするまでは、ほかのファイルにそのファイル番号を付けることはできません。

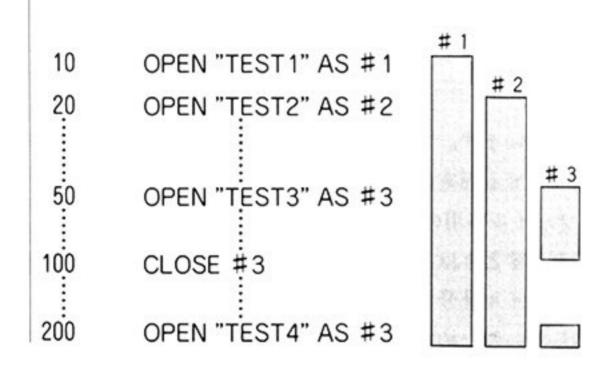
ファイル番号には、1からBASIC起動時の「How many files?（0-15）」で答えた数までの範囲を指定することができます。

5.2.4 同時にオープンするファイルの数

BASICは起動時に「How many files?(0-15)」と、同時にオープンするファイルの数を尋ねてきます。ファイルの入出力用のバッファの数は、ここで指定した数だけ用意します。数には最大15までの値を指定することができます。0から15以外の数を指定した場合、または［↵］キーだけを押した場合は、接続されているディスクドライブの数だけ設定します。

ここで設定した数は、再びBASICを立ち上げるまで変更できません。したがって、プログラム中で不足しないように、少し余裕のある数を設定するようにします。ただしバッファはユーザーエリアの中に確保されるため、不必要に多くの数を指定すると、それだけユーザーエリアが少なくなります。

「同時に」の意味：

「同時にオープンするファイルの数」と、プログラム中で使用するファイルの数は必ずしも同一ではありません。

例 プログラム中で使用するファイルはTEST1, TEST2, TEST3, TEST4の4つです。TEST1とTEST2は常に使用する必要がありTEST 3, TEST 4は必要なときだけ使用します。

この例では、4つのファイルのうち2つは同じファイル番号を使用することができます。したがって同時にオープンするファイルは3つで済みます。

5.3 シーケンシャルファイルとランダムファイル

5.3.1 シーケンシャルファイル

シーケンシャルファイルは書き込んだ順序でしかデータを読み出すことのできないファイルです。また既存のファイルに対して一部分のデータを直接、修正することができません。しかし書き込むデータの長さやデータ構成について特別な配慮が不要で、簡単にファイルを作成することができます。

(1) ステートメントと関数

シーケンシャルファイルに対して使用できるステートメントおよび関数は次のとおりです。

ステートメント・関数	説明
OPEN ～ INPUT	既存のファイルの先頭からデータの読み出しを可能にします。
OPEN ～ OUTPUT	ファイルを新規作成しデータの書き込みを可能にします。
OPEN ～ APPEND	既存のファイルの終わりから続けてデータを書き込みます。
CLOSE	ファイルをクローズします。データの書き込みを行った場合は必ず実行してください。
PRINT#	データを書き込みます。
PRINT# USING	書式指定をしてデータを書き込みます。
WRITE#	データを二重引用符(")とカンマ(,)とともに書き込みます。
INPUT#	データを読み出します。
LINE INPUT#	CRコードまでを1つのデータとして読み出します。
WIDTH	ファイルの1行の長さを設定します。
INPUT$	データを文字数を指定して読み出します。
LOC	ファイルの位置を256バイト単位で与えます。
LOF	ファイルの大きさを256バイト単位で与えます。
EOF	ファイルの終わりを検出します。
FPOS	ディスク上の読み書きしたセクタ位置を与えます。

(2) ファイルの構成

PRINT#文、PRINT# USING文、WRITE#文はそれぞれPRINT文、PRINT USING文、WRITE文によって画面に表示されるものと同じ形式でファイルにデータを書き込みます。

シーケンシャルファイルのデータとデータの間は次のような区切り記号によって区切る必要があります。

シーケンシャルファイルの区切り記号

数値データ	空白、カンマ(,)、CR コード
文字データ	カンマ(,)、CR コード

数値データ

```
100  OPEN "TEST.DAT" FOR OUTPUT AS #1
110  A=1234：B=−567.89
120  PRINT#1, A；B
130  CLOSE
140  '
150  OPEN "TEST.DAT" FOR INPUT AS #1
160  INPUT#1, A, B
170  PRINT A；B
180  CLOSE
190  END
```

120行を変えていくとファイルに書き込まれる形式はそれぞれ次のようになります。

```
120  PRINT#1, A；B
```

□ 1 2 3 4 □ − 5 6 7 . 8 9 □ CR LF EOF

□：空白　CR：&H0D　LF：&H0A　EOF：&H1A

数値データの後ろには空白が自動的に補われデータの区切りになります。

```
120  PRINT#1, A, B
```

これは PRINT 文と同じように14文字単位で書き込みます。

```
120  WRITE#1, A, B
```

1 2 3 4 , − 5 6 7 . 8 9 CR LF EOF

WRITE#文を使うとカンマ(,)でデータが区切られます。

文字データ

```
100 OPEN "TEST.DAT" FOR OUTPUT AS #1
110 A$="TOKYO" : B$="OSAKA"
120 PRINT #1, A$ ; B$
130 CLOSE
140 '
150 OPEN "TEST.DAT" FOR INPUT AS #1
160 INPUT#1, A$, B$
170 PRINT A$, B$
180 CLOSE
190 END
```

120行を変えていくとファイルに書き込む形式はそれぞれ次のようになります。

```
120 PRINT#1, A$ ; B$
```

[T][O][K][Y][O][O][S][A][K][A][CR][LF][EOF]

"TOKYO"と"OSAKA"に区切りがないために、INPUT#文で読み出すとA$に"TOKYOOSAKA"と一つのデータとして読み込み、B$に読み込むデータがなくなるためInput past end(読み出すデータがない)エラーになります。

```
120 PRINT#1, A$, B$
```

[T][O][K][Y][O][ ][ ][ ][ ][ ][ ][ ][ ][ ][O][S][A][K][A][CR][LF][EOF]

PRINT文と同じように14文字単位で書き込みますが文字データの場合、空白はデータの区切りにならないため"TOKYO　　　　　　　　　OSAKA"と一つのデータとして読み出し、Input past end(読み出すデータがない)エラーになります。

```
120 PRINT#1, A$ ; "," ; B$
```

[T][O][K][Y][O][,][O][S][A][K][A][CR][LF][EOF]

データとデータの間に区切り記号としてカンマ(,)を書き込み、"TOKYO"と"OSAKA"に分けて読み出すことができます。もしカンマやセミコロン(;)、先行する空白、改行、復改などを含んでいる場合はデータを二重引用符("")で囲んでください。

```
120  WRITE#1, A$, B$
```

" T O K Y O " , " O S A K A " CF LF EOF

自動的に二重引用符でデータを囲み、区切り記号としてカンマ(,)を挿入します。INPUT 文で読み出す際には二重引用符はデータとして読み出しません。

5.3.2 ランダムファイル

ランダムファイルは任意の場所のデータを直接に読み書きができるファイルです。シーケンシャルファイルは、データの長さが自由でしたが、ランダムファイルの場合はデータの長さを固定しなければなりません。

ランダムファイルで扱うことのできるデータの集まりの単位をレコードと呼びます。レコードはレコード番号により互いに区別することができます。またシーケンシャルファイルではデータはすべて文字形式で書き込みますが、ランダムファイルでは数値データはバイナリー形式で書き込みます。

(1) ステートメントと関数

ランダムファイルに対して使用されるステートメントおよび関数は次のとおりです。

ステートメント・関数	説明
OPEN	ファイルをランダムモードでオープンし入出力を可能にします。もしファイルが存在しない場合はファイルを作成します。
CLOSE	ファイルをクローズします。
FIELD	ランダムファイルバッファ(1レコードの大きさに相当)内の変数の割当の定義を行います。
LSET/RSET	データをランダムファイルバッファに書き込みます。
PUT#	ランダムファイルバッファのデータをファイルに書き込みます。(1レコードの書き込み)
GET#	ファイルから1レコードをランダムファイルバッファに読み出します。
MKI$/MKS$/MKD$	数値データを内部形式の文字列に変換します。MKI$ は整数を2バイトの文字列に、MKS$ は単精度実数を4バイトの文字列に、MKD$ は倍精度実数を8バイトの文字列に変換します。
CVI/CVS/CVD	文字列になっている数値データをもとの数値に変換します。CVI は2バイトの文字列を整数に、CVS は4バイトの文字列を単精度実数に、CVD は8バイトの文字列を倍精度実数に変換します。

ステートメント・関数	説明
FPOS	ディスク上の読み書きしたセクタ位置を与えます。
LOC	直前に入出力を行ったレコードのレコード番号を返します。
LOF	ランダムファイルの大きさを返します。

(2) ファイルの構成

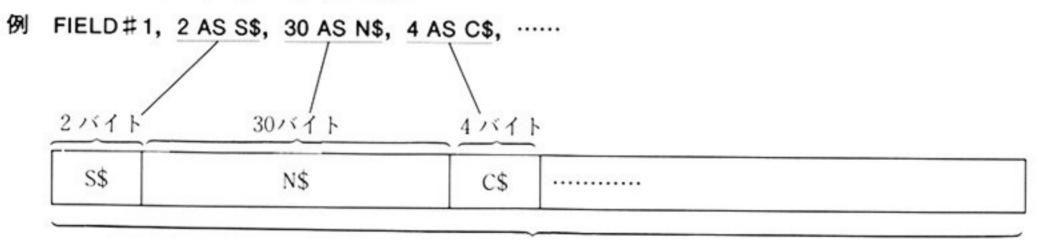
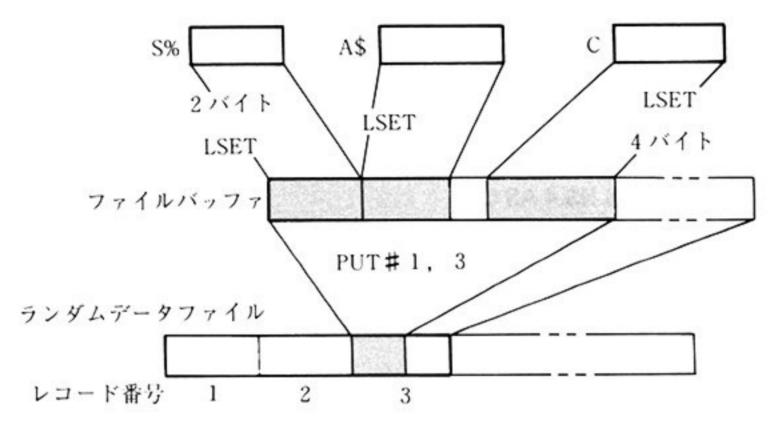
```
100 OPEN "TEST.DAT" AS ＃1
110 FIELD＃1,2 AS S$,30 AS N$,4 AS C$
120 INPUT "商品コード No.(0：終わり)";S%
130 IF S%=0 THEN 210
140 INPUT "商品名";A$
150 INPUT "価格";C!
160 LSET S$=MKI$(S%)
170 LSET N$=A$
180 LSET C$=MKS$(C!)
190 PUT＃1,S%
200 GOTO 120
210 CLOSE
```

レコードサイズ

ランダムファイルを構成する1レコードの長さは固定で、256バイトです。

ランダムファイルバッファ

ランダムファイルに対するレコードの入出力はランダムファイルバッファを介して行われます。このバッファに変数の割り当てを定義するのがFIELD文です。FIELD文で定義した変数の長さの合計は256バイト以下でなければなりません。

例　FIELD＃1, 2 AS S$, 30 AS N$, 4 AS C$, ……

数値はMKI$，MKS$，MKD$関数で文字列に変換してランダムファイルバッファへ書き込みます。ランダムファイルバッファへの書き込みはLSET文、RSET文を用いなければなりません。通常の代入文を用いるとFIELD文で定義したランダムファイルバッファ用の変数が一般の変数とみなされ、ランダムファイルバッファへの書き込みが行われなくなってしまいます。LSET文は変数に対して左詰め、RSET文は右詰めでデータをセットします。いづれの場合も余りの部分には空白をセットします。

PUT#文を使ってファイルにデータを書き込みます。PUT#文で指定するレコード番号は1から32000までです。省略すると直前にPUT#文で書き込んだレコードの次の番号に書き込みます。

GET#文によりランダムファイルからデータを読み出します。省略すると直前にGET#文で読み出したレコードのデータを読み出します。

5.4 入出力装置に対するステートメントと関数

BASIC は種々の入出力装置をすべてファイルという共通の考え方で扱っています。したがって入出力装置の違いを意識することなく BASIC のステートメント、関数を使用することができます。しかし入出力装置によっては入力のみであったり出力のみであったりします。この違いとステートメントと関数の関係をまとめたのが次の表です。

デバイス名 / ステートメント・関数	KYBD：	SCRN：	LPT1：	COMn：	n：(ディスク)
BLOAD	×	×	×	○	○
BSAVE	×	×	×	○	○
CHAIN	○	×	×	○	○
CLOSE	△	△	○	○	○
DSKF	×	×	×	×	○
EOF	×	×	×	○	○
FPOS	×	×	○	×	○
GET#	○	×	×	×	○
INPUT#	○	×	×	○	○
INPUT$	○	×	×	○	○
LINE INPUT#	○	×	×	○	○
LOAD	○	×	×	○	○
LOC	○	×	×	○	○
LOF	×	×	×	○	○
MERGE	×	×	×	×	○
OPEN(入力)	○	×	×	○	○
OPEN(出力)	×	○	○	○	○
OPEN(ランダム)	×	×	×	×	○
OPEN(追加)	×	×	×	×	○
PRINT#	×	○	○	○	○
PRINT USING#	×	○	○	○	○
PUT#	×	○	○	×	○
SAVE	×	○	○	○	○
WIDTH	×	×	○	○	○
WRITE#	×	○	○	○	○

○：使用できる

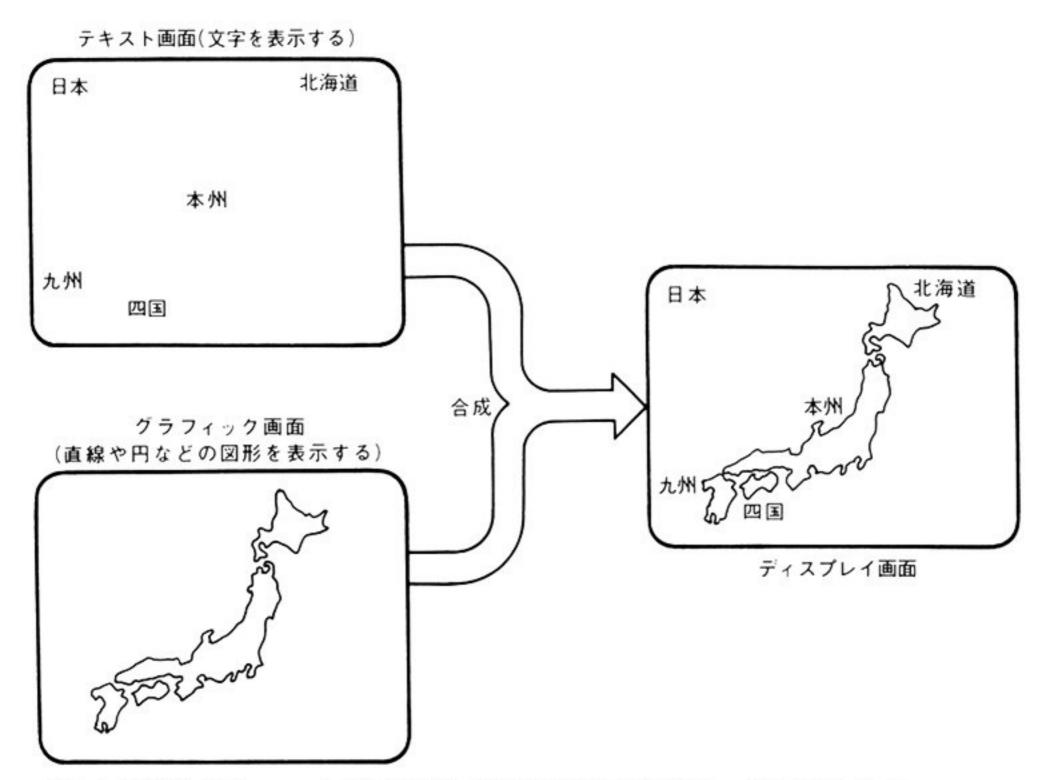
×：使用できない

△：使用できるが、データの入出力は行わない

テキスト画面とグラフィック画面

BASICは文字を表示するテキスト画面と、グラフィックを表示するグラフィック画面を持ち、それぞれにいろいろな画面モードがあります。この章ではテキスト画面とグラフィック画面の各モードおよびこれらの画面と表示装置の関係などを説明します。

6.1 テキスト画面とグラフィック画面

BASICは文字を表示するテキスト画面とグラフィックを表示するグラフィック画面をそれぞれ別に持っており、ディスプレイ装置にはこれらを重ね合わせたものを表示します。したがって、それぞれの画面に対する命令は独自になります。

テキスト画面とグラフィックス画面はそれぞれ別の操作で画面のモードを設定します。

6.2 テキスト画面

6.2.1 画面モード

テキスト画面は文字を表示するための画面です。テキスト画面は表示できる文字数の違いにより次のモードに分けられます。

テキスト画面モード	ディップスイッチ SW2	
	3	4
80桁25行モード	ON	ON
80桁20行モード	ON	OFF
40桁25行モード	OFF	ON
40桁20行モード	OFF	OFF

ここでの桁は英文字や数字などの1バイト文字(半角文字)分を1桁とします。したがって、漢字やひらがななどの2バイト文字(全角文字)は2桁分必要になります。

BASIC 起動時のテキスト画面のモードは、本体のディップスイッチ SW2 の設定によって決まります。

プログラム中でモードを切り換えるには WIDTH 文を使用します。画面モードを切り換えると自動的にテキスト画面を消去(表示しているすべての文字が消える)します。

テキスト画面では表示できる文字数のほかに、さらに次の3つの条件を変えることができます。

- スクロール領域の設定
- ファンクションキーの表示
- カラー/白黒モードの設定

これらの設定は、すべて CONSOLE 文を使用して変更します。

CONSOLE [スクロール開始行], [スクロール行数], [ファンクションキー表示], [白黒/カラー]

0 1 2…　…79
0
1
2
23
24
この領域の表示は変化しません。
スクロール領域
(表示が変化します。)

スクロール領域はテキストを表示する領域を指定するもので、表示するデータがスクロール範囲の下段右下に行くと1行表示が上に上がります。これをスクロールといいます。

スクロール範囲の外を固定表示部といい、ここはスクロールの影響を受けません。固定表示部に文字を表示する場合は、LOCATE文でカーソルを移動します。

ファンクションキー表示は、ファンクションキーに設定された文字を、画面の最下行に表示するかしないかを指定するものです。

白黒/カラースイッチはテキスト画面に表示する文字を白黒で表示するか、カラーで表示するかを決定するものです。ここでカラーの設定をすることによりCOLOR文で8色のカラー表示を行うことができるようになります。

6.2.2 文字の種類

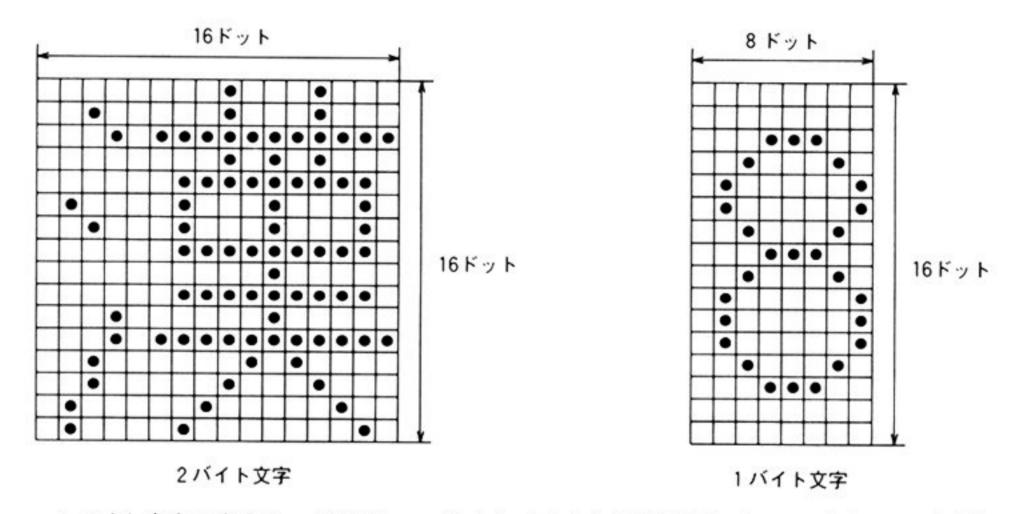
テキスト画面に表示する文字には、次の２種類があります。

１バイト文字(半角文字とも呼ばれる)

２バイト文字(全角文字とも呼ばれる)

２バイト文字はちょうど１バイト文字２文字分の表示領域を必要とします。したがって表示できる文字数は半角文字の半分になります。

注意 │２バイト文字のうち、&H2920 から &H2B7E までの文字は、表示は半角文字になります。

２バイト文字　　　　　１バイト文字

１バイト文字は文字コード(内部コード)として０から255(16進数で &H00 から &HFF)を持つ256文字から構成されます。ただし、この中で実際に画面上に文字として表示するのは216文字で、それ以外は、特に制御文字と呼び、画面や周辺機器に対して特別な動作を行います。

例

PRINT CHR$(&H41)	文字コードとして &H41 を持つ文字(A)を画面に表示します。
A	(PRINT "A"としても同じ結果を得ます。)
OK	
PRINT CHR$(7)	文字コードとして７を持つ文字を画面に表示します。ブザーを鳴らす制御文字のため画面には何も表示されません。
OK	

２バイト文字は文字コードとして &H2121 から &H7C7E を持つ漢字やひらがなです。漢字コードについてはコンピュータ本体のユーザーズマニュアルを参照してください。

ユーザー定義文字

BASICではJIS第1水準、第2水準の漢字のほかに、ユーザーが自由に作成した文字を2バイト文字の一部として扱えるようになっています。これらの文字はユーザー定義文字とよび、JIS漢字コードの&H7621～&H767Eおよび&H7721～&H777Eに割り当てられます。

ユーザー定義文字はユーティリティソフトBMENUの「ユーザー・フォント・メンテナンス」により簡単に作成することができます。

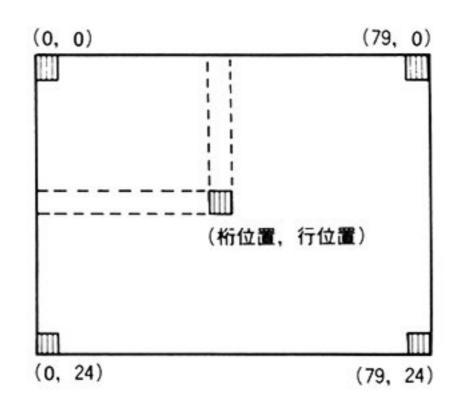
BASIC起動時に、システムディスクにユーザー定義文字を格納したファイル「userfo. nt」が存在すると、自動的にこのファイルを読み込みます。この場合ユーザー定義文字に相当する漢字コードを指定すると、ここで読み込んだファイルの文字パターンを表示します。

このほかにBASICの命令としてKPLOAD文があります。この命令を使えばプログラム中で自由にユーザー定義文字の設定を行うことができます。

注意 ユーザー定義文字はプリンタにそのまま打ち出すことはできません。ユーザー定義文字をプリンタに打ち出す場合は、プリンタの制御コードに従ってプリンタにプリンタのユーザー定義文字を設定する必要があります。

6.2.3 キャラクタ座標

テキスト画面に設定されている座標系をキャタクタ座標といい(桁位置,行位置)で示します。画面の左上の文字の場所が(0, 0)、右下の文字が(1行の桁数－1, 画面の行数－1)になります。例えば80桁25行モードの場合は次のようになります。

例 80桁25行モード

6.2.4 カラー指定

表示する文字の色はCOLOR文で指定します。ただし、この設定はCONSOLE文でカラーモードになっていなければなりません。

カラーモードでない場合は、それぞれの文字に特別な表示を行わせることができます。

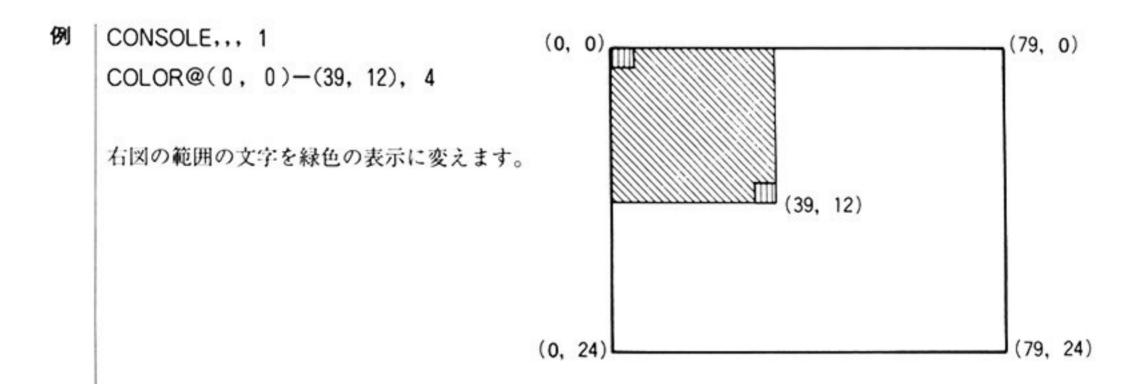
COLOR [色指定], [背景色], [ボーダーカラー], [前景色], [拡張カラー指定]

値＼モード	意味	
	カラーモード	白黒モード
0	黒	ノーマル(通常表示)
1	青	ヒドン(文字を表示しない)
2	赤	ブリンク(点滅表示)
3	紫	ヒドン
4	緑	リバース(反転表示)
5	水色	リバースヒドン(反転表示で文字を表示しない)
6	黄色	リバースブリンク(反転の点滅表示)
7	白	リバースヒドン

背景色、前景色、拡張カラーモードはグラフィック画面に対して働く機能です。

このほか、すでに画面上に表示している文字の色を変えることができます。

COLOR@(X1, Y2)−(X2, Y2), 色指定

キャラクタ座標(X1, Y1), (X2, Y2)で囲まれた範囲に表示している文字を、指定した色で表示し直します。色指定はカラーモード、白黒モードで異なりCOLOR文と同じです。この設定はすでに表示している文字のみに有効ですので、この命令実行後、指定範囲に表示した文字はなんの影響も受けません。COLOR文で指定した色で表示します。

6.3 グラフィック画面

6.3.1 画面モード

グラフィック画面はグラフィックを表示するための画面です。グラフィック画面には次の機能を組み合わせて、さまざまな画面モードを設定できます。

	機　能	意　味	方　法
①	白黒/カラースイッチ	グラフィック画面でのカラー/白黒モードの切り換え	SCREEN 文
②	標準/高解像スイッチ	グラフィック画面の精度を決める	
③	8色/4096色スイッチ	カラーモードで表示する色の数を決める	ディップスイッチ SW1-8 COLOR 文

①、②についてはSCREEN 文でプログラム中のどこでも変更することができますが、③については本体のディップスイッチ SW1-8の設定によりBASIC 起動時に決まります。また拡張グラフィックモードにすると、32KB の16色表示のためのシステムがメモリに読み込まれるためユーザーエリアが減ります。

	機　能	意　味	
SW1-8	グラフィックモードの設定	OFF	8色中8色モードのみ使用(基本グラフィックモード)
		ON	4096色モード使用可能(拡張グラフィックモード)

グラフィック画面のモードは、標準カラーモード、標準白黒モード、高解像白黒モード、高解像カラーモードの4つです。画面モードはSCREEN 文で設定します。

SCREEN [画面モード] [, 画面スイッチ] [, 描画ページ] [, 表示ページ]

それぞれのパラメータの意味は次のとおりです。

(1) 画面モード

0から3までの数値で、画面モードを決めます。画面モードにより画面の精度、使用できる画面数が違います。また、画面数はディップスイッチSW1-8の指定によっても異なります。グラフィック画面は、画面モードに応じて同じ大きさの画面を複数枚持っています。これらを効果的に使用することにより高速な描画を行わせることもできます。

指定値	画面精度	色	モード名	画面数
0	標準モード（横640×縦200ドット）	カラー	標準カラーモード	4
1		白黒	標準白黒モード	16(12)
2	高解像モード（横640×縦400ドット）	白黒	高解像白黒モード	8（6）
3		カラー	高解像カラーモード	2

（　）内の値は基本グラフィックモードの場合です。

ただし使用するディスプレイによっては指定できないモードもあります。次表に使用ディスプレイの種類による画面モードを示します。

ディスプレイ ＼ モード		標準カラーモード	標準白黒モード	高解像カラーモード	高解像白黒モード
高解像ディスプレイ	カラー	○	○	○	○
	モノクロ	△	○	△	○
標準ディスプレイ	カラー	○	○	×	×
	モノクロ	△	○	×	×

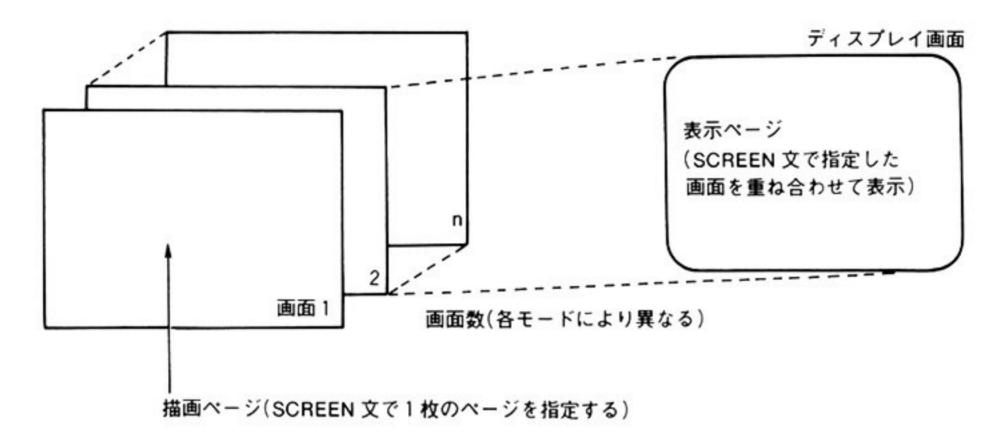
○：指定可　×：指定不可　△：指定はできるが単色の色の濃淡のみ

(2) 画面スイッチ

グラフィック画面に表示している内容を一時的に消します。0から3までの数値で指定します。

指定する値	機　　能	
0	ディスプレイにグラフィック画面のデータを表示します。	フラッシュレス（標準）描画
1		フラッシュ（高速）描画
2	ディスプレイにグラフィック画面のデータを表示しません。	フラッシュレス（標準）描画
3		フラッシュ（高速）描画

(3) 描画ページ

描画ページと表示ページの関係は次のようになります。

このため描画ページと表示ページが違った場合は書き込んだデータがディスプレイには表示されません。ただし表示ページを変えれば表示することができます。

描画ページはグラフィック命令を実行する画面を指定します。画面モードによっても指定できる値が違います。

モード	指定する値
標準カラーモード	0～3
標準白黒モード	0～15(0～11)
高解像カラーモード	0～1
高解像白黒モード	0～7(0～5)

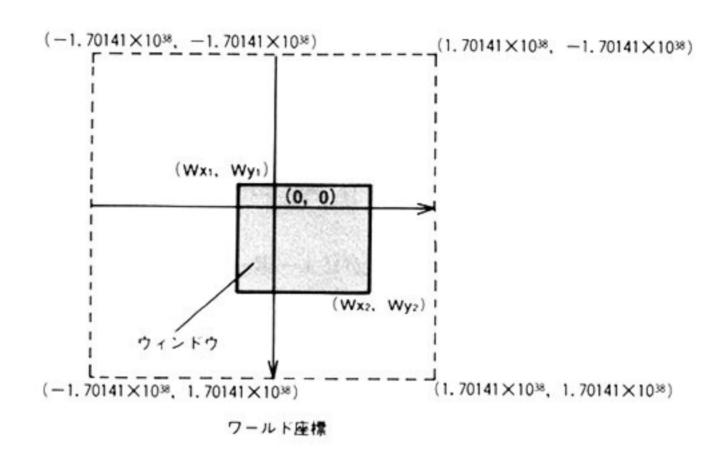
指定した値＋1が実際の画面のページ番号になります。(　)の値は基本グラフィックモードの場合です。

(4) 表示ページ

ディスプレイに表示する画面のページを指定します。指定する数値がそのままページ番号を表すわけではありません。画面モードによって指定できる値と意味が違います。また、標準白黒モードおよび高解像白黒モードでは、3つまでのページを重ね合わせてディスプレイの画面に表示することができます。表示ページの値によってどのページを重ね合わせて、ディスプレイに表示されるかが決ります。指定する値と実際の表示ページの関係は「日本語 Disk BASIC リファレンスマニュアル」の SCREEN 文の項を参照してください。

6.3.2 ワールド座標とウィンドウ

グラフィック画面は架空の論理的な大きさの座標をもっており、この画面上にグラフィックデータを描きます。

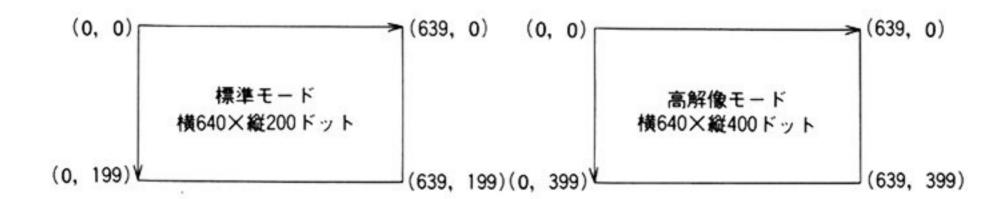
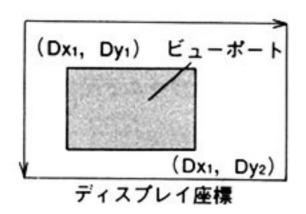
この大きな画面に設定しているのがワールド座標と呼ばれる座標系です。ワールド座標は論理的に考えた架空の大きさの座標系です。座標の範囲はX方向(横方向)、Y方向(縦方向)とも-1.70141×10^{38}から$+1.70141\times10^{38}$です。ほとんどのグラフィック関係の命令はワールド座標の中にグラフィックを描きます。

ワールド座標

ワールド座標系は、架空のもので、実際に画面上で見ることのできる範囲は限られています。画面上に表示する範囲をウィンドウといいます。ウィンドウの大きさはWINDOW文で設定します。

WINDOW(Wx_1, Wy_1)−(Wx_2, Wy_2)

(Wx_1, Wy_1)と(Wx_2, Wy_2)を対角とするワールド座標上の領域がウィンドウになります。

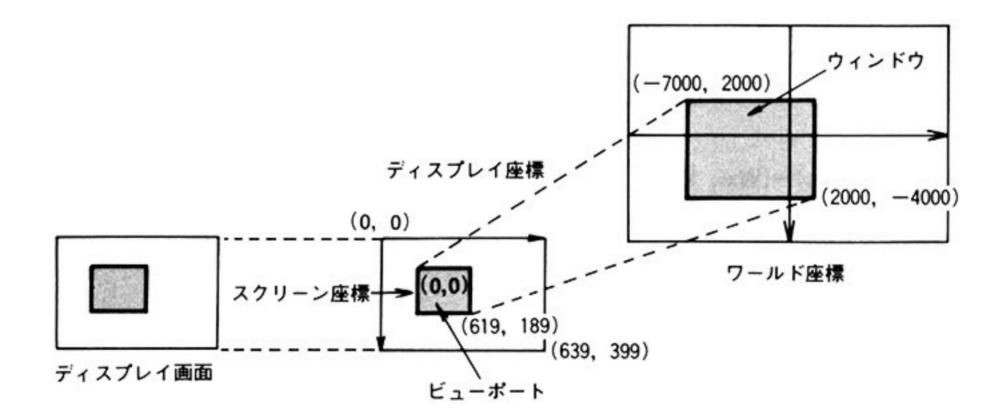
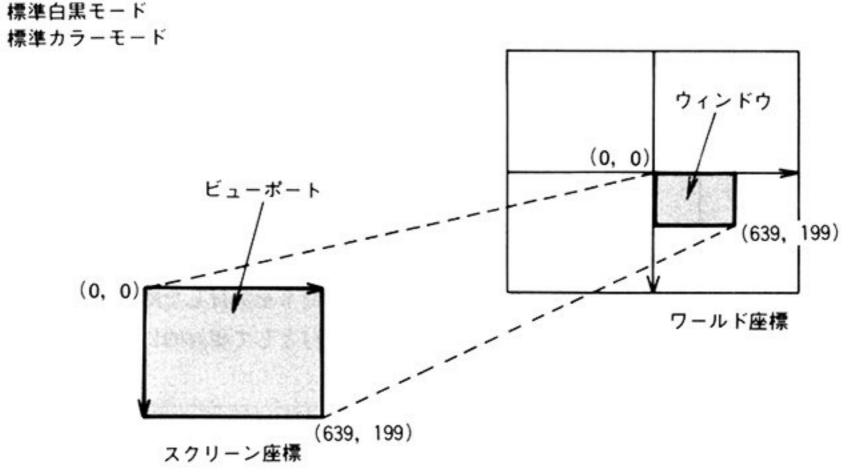
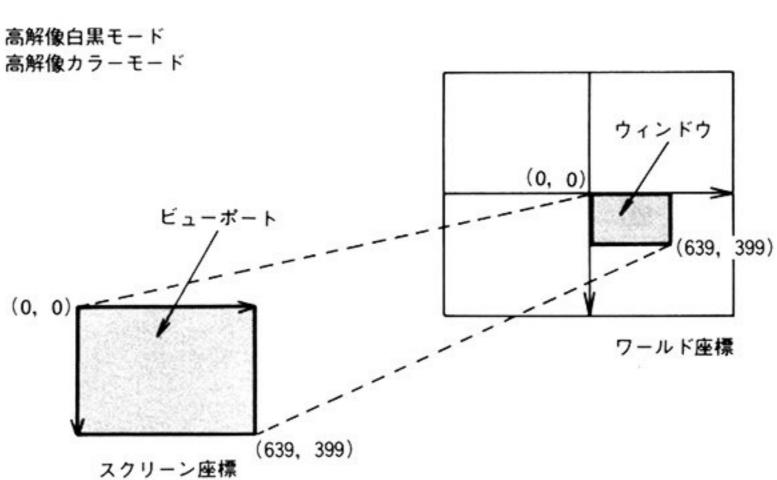
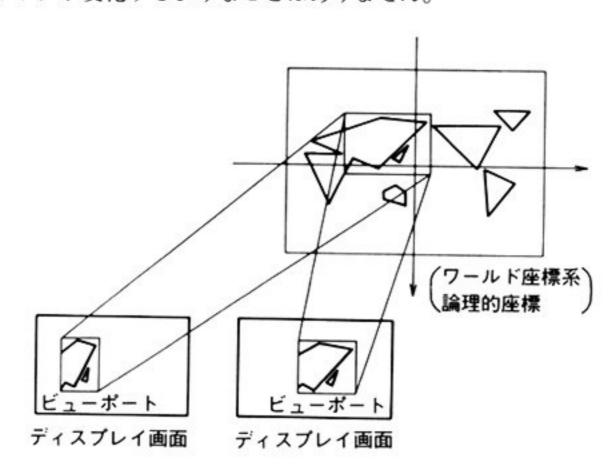
6.3.3 ディスプレイ座標とビューポート

ディスプレイ座標は、画面の精度により物理的に設定されている座標系で次のようになります。高解像モードでは(0, 0)から(639, 399)の範囲、標準モードでは(0, 0)から(639, 199)の範囲になります。

このディスプレイ座標に実際のウィンドウのデータを表示する範囲をビューポートといいます。ビューポートの位置と大きさはVIEW文で設定します。

VIEW(Dx1, Dy1)−(Dx2, Dy2) [, 領域色] [, 境界色]

(Dx1, Dy1)と(Dx2, Dy2)を対角とする領域がビューポートになります。

6.3.4 スクリーン座標

画面上のビューポートに設定されるのがスクリーン座標です。スクリーン座標はビューポートの左上隅の点(ドット)を(0, 0)とした座標系です。

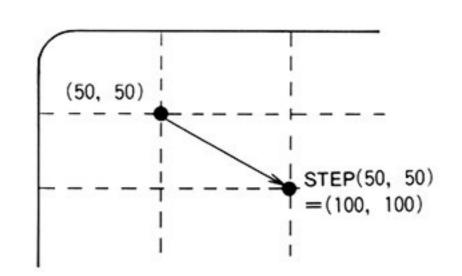
以上をまとめると座標の関係は次のようになります。

BASIC 起動時のウィンドウ、ビューポートの関係は次のようになっています。

ウィンドウやビューポートを変えることによって、グラフィックの移動・拡大・縮小を行うことができます。ただしウィンドウやビューポートの設定は、すでに表示されているグラフィックに対しては働きません。したがってウィンドウやビューポートの設定を変更することによって、ただちに表示されているグラフィックが変化するようなことはありません。

6.3.5　座標指定の形式

グラフィック関係の命令を実行するのには、グラフィック画面上のどの位置(座標)に表示するのかを指定します。その際の座標の指定方法には、次の2つの形式があります。

絶対座標形式：(X, Y)	グラフィック画面の実際の位置を指定。
相対座標形式：STEP (X, Y)	最終参照座標からの位置で指定 最終参照座標はグラフィック関係の命令を実行した際に最後に指定した点のことをいいます。この点を(0, 0)として相対的に指定するのが相対形式になります。

例　次の場合、同じ位置に点を表示します。

```
① PSET(50, 50)
   PSET(100, 100)

② PSET(50, 50)
   PSET STEP(50, 50)
```

次の例では同じ位置に四角形を描きます。

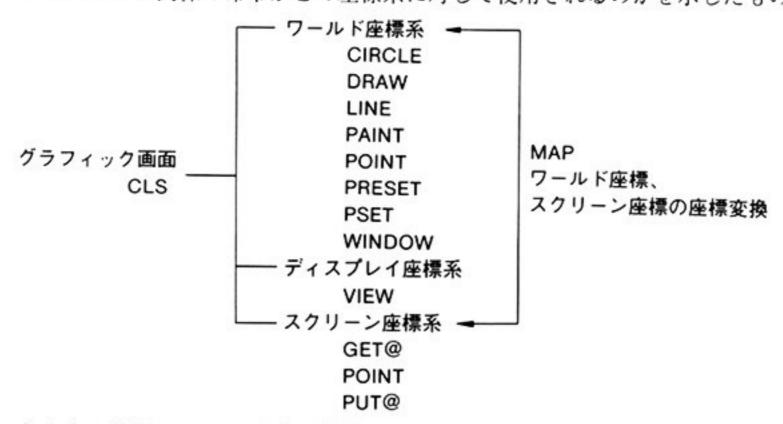
例

```
① LINE(100, 100)-(200, 200),, B

② LINE(100, 100)-STEP(100, 100),, B
```

最終参照座標はビューポートやウィンドウを設定したときやCLS文を実行したときにスクリーン座標の(0, 0)の位置に初期設定されます。

6.3.6　座標系の整理

グラフィック関係の命令がどの座標系に対して使用されるのかを示したものが次の図になります。

各命令の詳細については「日本語Disk BASICリファレンスマニュアル」を参照してください。

6.3.7 カラー表示

グラフィック画面をカラーモードにするためにはCOLOR文を使用します。

COLOR [文字の色],[背景色],[境界色],[前景色],[カラーモード]

文字の色はテキスト画面の文字の色を指定するものです。グラフィック画面には直接関係ありません。

背景色はバックグランドカラーともいい、グラフィック画面の背景の色を設定するものです。この設定を行った後、CLS文を実行すると、この色でグラフィック画面を塗りつぶします。

境界色は表示装置の画面上でBASICによって扱えない外枠の部分の色です。標準カラーディスプレイに対してのみ意味を持ちます。

前景色はグラフィック命令で特に色指定がない場合にグラフィックを描くのに使う色です。

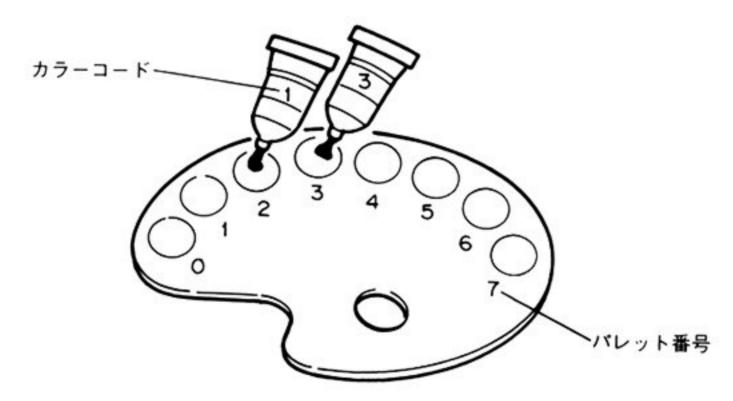
カラーモードはBASICで使用できる色の種類を0から2の値で設定します。

0	固定した8色のカラー表示を行います。このモードを8色中8色モードといいます。
1	4096色中の8色のカラー表示を行います。このモードを4096色中8色モードといいます。
2	4096色中の16色のカラー表示を行います。このモードを4096色中16色モードといいます。

グラフィック画面で、色を指定する場合は直接、色の種類を指定するのではなく間接的なパレット番号を指定します。パレットは16色モードのときに0から15までの16個、8色モードのときに0から7までの8個あり、それぞれに色を割り当てます。パレット番号に割り当てられる色を変えることにより瞬時に表示している色を変更することができます。

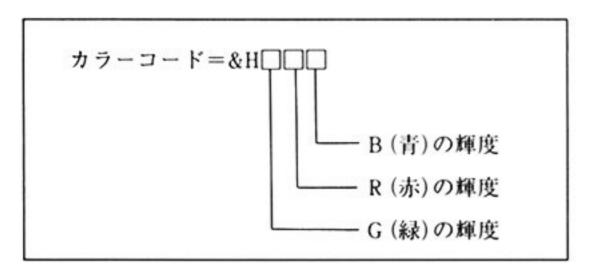
各パレットに実際に色を対応づけるには**COLOR=**文を使います。

COLOR [=(パレット番号,カラーコード)]

=以下を省略すると色の設定は初期設定になります。カラーコードはカラーモードにより異なります。

8色中8色モードのカラーコードは次のようになります。

カラーコード	0	1	2	3	4	5	6	7
色	黒	青	赤	紫	緑	水色	黄色	白

4096色中８色モード、4096色中16色モードの場合は次のような形式でカラーコードを示します。

この場合のカラーコードは光の３原色のＧ（緑）、Ｒ（赤）、Ｂ（青）の混ぜ合わせによって指定します。下図はその様子を示したものです。

中間色は、輝度を調整することで表現できます。８色中８色モードはこの３つの色を同じ割合でしか混ぜ合わせることができません。

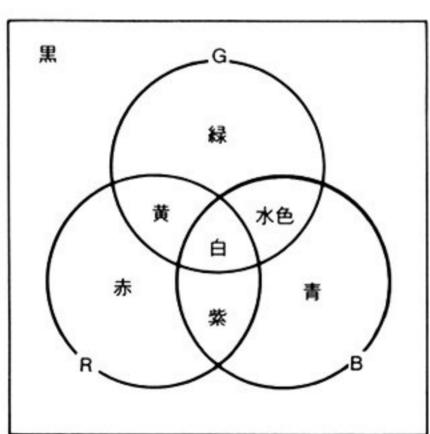
絵の具の場合、中間色を表現するためには混ぜ合わせる絵の具の量を調整します。これと同じようにディスプレイの場合には、色を混ぜ合わせる割合を輝度（色の明るさ）を調整することによって表現します。例えばオレンジ色は緑と、この緑より明るい赤を混ぜ合わせることによって作ることができます。

Ｇ（緑）、Ｒ（赤）、Ｂ（青）はそれぞれ16段階の輝度を表現することができます。したがって３色、16段階の輝度を自由に組み合わせることによって4096色を表現することができます。

4096色＝（Ｇの16輝度）×（Ｒの16輝度）×（Ｂの16輝度）

それぞれの色は4ビットのデータで16段階の輝度を表し、0からFまでの16進数で表現します。カラーコードの上位４ビットに緑、中位４ビットに赤、下位４ビットに青の輝度を指定します。

輝度が０とはその色を「使わない」ことを意味し、輝度がＦとは、その色の「最も明るい色」を使うことになります。

輝度	0 ・・・・・・・・・・・・・・・・・・・・・・・・・・・・ F
	暗い　　　　　　　　　　　　　　　　　　　　明るい

例

オレンジ色＝（G：&H9）＋（R：&HF）＋（B：0）
　　　　　＝&H9F0
明るい黄色＝（G：&HF）＋（R：&HF）＋（B：0）
　　　　　＝&HFF0
暗い黄色＝（G：&H7）＋（R：&H7）＋（B：0）
　　　　　＝&H770

カラーコードとパレットの初期設定は次のとおりです。これはBASIC起動時、カラーモードを指定した**COLOR**[1]文およびパレット番号、カラーコードを省略した**COLOR**[2]文実行時に設定される値です。

8色中8色モード

パレット番号	カラーコード
0	0(黒)
1	1(青)
2	2(赤)
3	3(紫)
4	4(緑)
5	5(水色)
6	6(黄色)
7	7(白)

4096色中8色モード

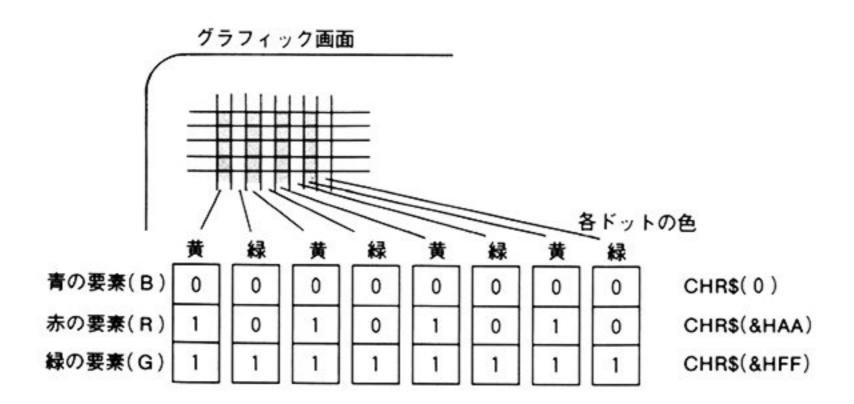
パレット番号	カラーコード
0	&H000(黒)
1	&H00F(青)
2	&H0F0(赤)
3	&H0FF(紫)
4	&HF00(緑)
5	&HF0F(水色)
6	&HFF0(黄色)
7	&HFFF(白)

4096色中16色モード

パレット番号	カラーコード	
0	&H000(黒)	
1	&H00F(青)	明るい
2	&H0F0(赤)	明るい
3	&H0FF(紫)	明るい
4	&HF00(緑)	明るい
5	&HF0F(水色)	明るい
6	&HFF0(黄色)	明るい
7	&HFFF(白)	明るい
8	&H777(灰色)	
9	&H00A(青)	暗い
10	&H0A0(赤)	暗い
11	&H0AA(紫)	暗い
12	&HA00(緑)	暗い
13	&HA0A(水色)	暗い
14	&HAA0(黄色)	暗い
15	&HAAA(白)	暗い

6.3.8 タイリング

タイリングは横８ドット縦ｎドットの矩形領域(タイルといいます)に指定した模様で指定領域内を塗りつぶすテクニックです。画面上のドットの大きさが小さいため各ドットに違う色を設定するとタイルの色があたかも割り当てた色の混合色であるかのように見えます。これにより８色中８色モードでも中間色を表現することができます。

例えば、横８ドットの各色の要素を次のように割り当てれば、この横８ドットのタイルは黄緑色に見えます。１はその要素が有り、０は無いことを表します。

このテクニックを使えば、８色中・８色モードでもいろんな色を表示することができます。

タイリングを行うにはPAINT文、CIRCLE文、LINE文などを使います。詳しくは、「日本語Disk BASICリファレンスマニュアル」のPAINT文の説明を参照してください。

例　次のプログラムは矩形を黄緑色で塗り潰します。

```
10 SCREEN 3, 0
20 CLS 2
30 TILE$=CHR$(&H00)+CHR$(&HAA)+CHR$(&HFF)
40 LINE(30, 30)-(90, 150),, BF, TILE$
```

第7章

プリンタ

本章では、プリンタに印字する方法と画面のハードコピーについて説明します。

7.1 プリンタへの印字

プリンタに印字する場合、使用するプリンタや使用する機能に応じて次のメモリスイッチの値を変更します。

メモリスイッチ

	機能	意味	
SW5-0	使用するプリンタ	0	PC-PR201 シリーズ以外のプリンタ
		1	PC-PR201 シリーズのプリンタ

	機能	意味	
SW6-4	画面ハードコピーの拡張	0	拡張しない
		1	拡張する

	機能	意味	
SW5-3	カラーハードコピー	0	白黒
		1	カラーコピー

メモリスイッチ SW6-2 で画面ハードコピー機能の拡張を指定するとカラープリンタでカラーハードコピーを行うことができます。

7.2 画面ハードコピー

画面ハードコピーは画面に表示している文字やグラフィックを、そのままプリンタに出力する機能です。画面ハードコピーは COPY キーによる方法と COPY コマンドによる方法があります。ハードコピーはメモリスイッチの内容により機能が異なります。

メモリスイッチ SW6-4 が 0 の場合

キー操作	機　能	対応するコマンド
COPY	テキスト画面とグラフィック画面を重ね合わせたハードコピーを行います。テキスト画面の文字はプリンタの字体になります。	COPY COPY 3
CTRL + COPY	テキスト画面のみのハードコピーを行います。	COPY 1
GRPH + COPY	グラフィック画面のみのハードコピーを行います。	COPY 2
―	標準モードの場合にグラフィック画面を縦方向に縮小してハードコピーを行います。	COPY 4
―	標準モードの場合にグラフィック画面を縦方向に縮小して、テキスト画面を重ね合わせてハードコピーを行います。	COPY 5

高解像モードの場合に COPY 4，COPY 5 を実行した場合は、それぞれ COPY 2，COPY 3 と同じになります。

メモリスイッチ SW6-4 が 1（画面ハードコピーの拡張）の場合

キー操作	機　能	対応するコマンド
COPY	テキスト画面とグラフィック画面を重ね合わせたハードコピーを行います。テキスト画面の文字はビットイメージで出力します。カラーコピーの場合は白黒を反転して出力します。*	COPY COPY 3
CTRL + COPY	テキスト画面のみのハードコピーを行います。	COPY 1
GRPH + COPY	グラフィック画面のみのハードコピーを行います。カラーコピーの場合は白黒を反転して出力します。	COPY 2
―	グラフィック画面のみのハードコピーを行います。白黒を反転しません。	COPY 4
―	テキスト画面とグラフィック画面を重ね合わせたハードコピーを行います。テキスト画面の文字はビットイメージで出力します。白黒を反転しません。*	COPY 5

カラーコピーはメモリスイッチ SW5-3を 1 にします。ただし4096色中 8 色モードおよび16色モードのときはカラーコピーを指定しても白黒コピーになります。

* テキスト画面が20行モードで、文字を反転表示している場合は画面のイメージとプリンタの出力結果が異なります。

7.3 ページプリンタ

ページプリンタを使用している場合、BMENUによりページプリンタの機能を使用した画面ハードコピーを行うことができます。

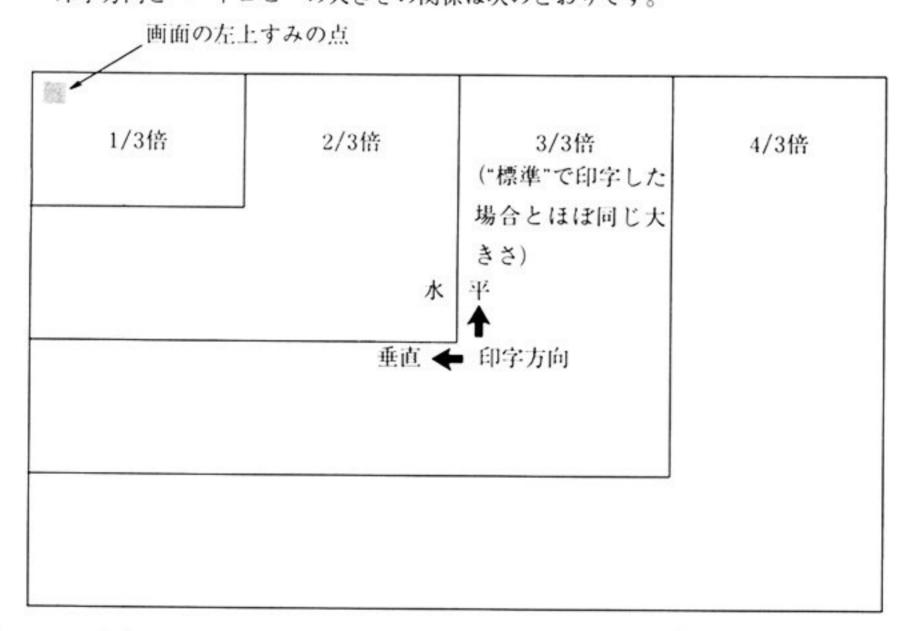
ハードコピー機能の拡張方法

① BMENUの「拡張ハードコピー･オートスタート設定」の中で、「拡張ハードコピー環境設定」を選びます。

② COPYコマンドとCOPYキーの機能を次のような設定に変更することができます。

キー操作	対応するコマンド	印字方向	ハードコピーの大きさ
COPY	COPY COPY 3	水平, 垂直	標準, 1/3, 2/3, 3/3, 4/3
CTRL + COPY	COPY 1		
GRPH + COPY	COPY 2		
―――	COPY 4	機能を変更することはできません。	
―――	COPY 5		

それぞれのキー操作(COPYコマンド)に対して、別々に印字方向とハードコピーの大きさを設定することができます。"標準"はページプリンタの機能を使わないでハードコピーを行う(一般のプリンタ用)場合に設定します。ページプリンタを使用していない場合は、必ず"標準"に設定にしてください。他の設定になっていると正常にハードコピーが行えません。"標準"設定の場合は水平/垂直の設定は無意味です。初期設定は"標準"になっています。

印字方向とハードコピーの大きさの関係は次のとおりです。

③ ここで設定した機能は、再度BASICを起動したときから有効になります。

注意 | ページプリンタはPC-PR601系と同等の機能を持ったページプリンタのことです。

第8章

データ通信

本章ではRS-232Cインターフェイスを用いたデータ通信の方法について説明します。

8.1 通信パラメータ

通信回線(RS-232Cインターフェイス)を用いたデータ通信は、ファイルの概念で扱うことができます。ただしディスクとは異なり、ファイル名の代わりに通信のパラメータを指定します。通信パラメータは、データ通信を行う相手(モデムやコンピュータ)と一致させる必要があります。これらのパラメータは本体のメモリスイッチを用いて設定しますが、一部はOPEN文のパラメータとしても設定できます。メモリスイッチを変更した後はディップスイッチSW2-5をONにしてリセットしてください。

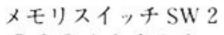
OPEN文で設定する場合はデバイス名に続いて通信パラメータを指定します。

```
"COM[回線番号]:(pcsxh)"
                p…パリティビット
                c…データビット長
                s…ストップビット
                x…XON/XOFF制御
                h…SI/SO制御
```

メモリスイッチSW1

7 6 5 4 3 2 1 0

機　能	意　　味			OPEN文のパラメータ
XON／XOFF制御	0	XON／XOFF制御を行わない		N
	1	XON／XOFF制御を行う		X
データビット長	1	0	7ビット	7
	1	1	8ビット	8
パリティチェック	*	0	パリティビットなし	N
	0	1	奇数パリティ	O
	1	1	偶数パリティ	E
ストップビット長	0	1	1ビット	1
	1	0	1.5ビット	2
	1	1	2ビット	3

*は0または1のどちらでもよい。

機　能	意　　味					OPEN文のパラメータ
通信速度	0	0	0	1	75　bps	OPEN文では設定不可
	0	0	1	0	150　bps	
	0	0	1	1	300　bps	
	0	1	0	0	600　bps	
	0	1	0	1	1200bps	
	0	1	1	0	2400bps	
	0	1	1	1	4800bps	
	1	0	0	0	9600bps	
SI／SO制御	0	SI／SO制御行わない				N
	1	SI／SO制御行う				S

通信速度

データを送信する速度です。

データビット長

1バイトのデータの長さを7ビットあるいは8ビットに設定します。

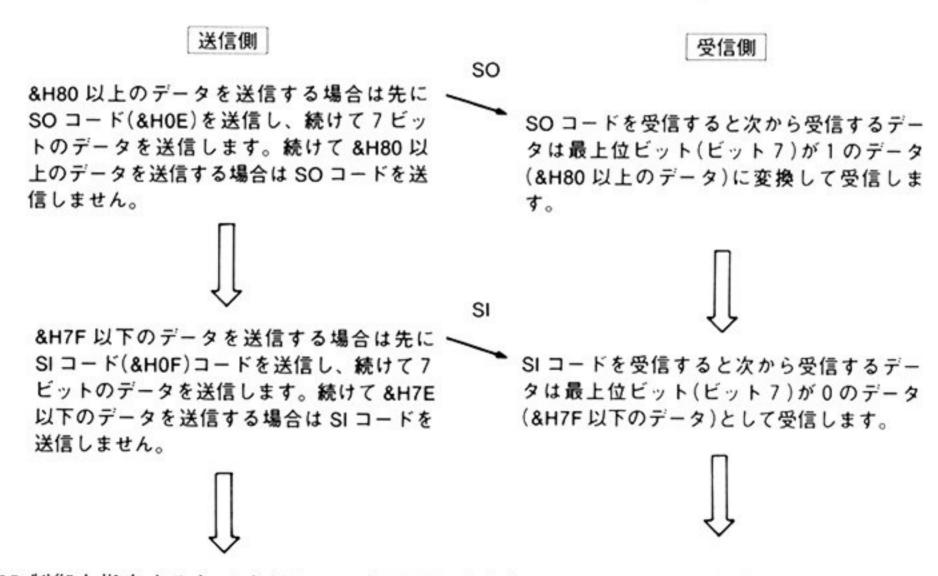
パリティビット

データを正確に送信できたかを確認するために、パリティビットと呼ばれる1ビットのデータを付加することができます。パリティビットを「あり」にすると、送信データに自動的にパリティビットを付加し、受信データについてはパリティチェックを行います。パリティビットには2種類あり、データビットの1のビット数を数え1の数を偶数になるように設定するのが偶数パリティ、奇数になるようにするのが奇数パリティです。

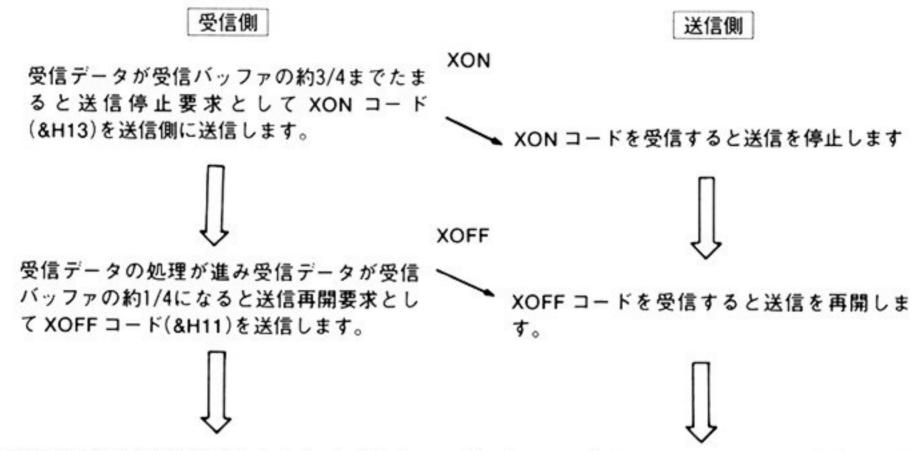
ストップビット

ストップビットは1バイトデータの終わりを示します。

SI/SOデータ制御

データビットを7ビットにすると16進数で&H00から&H7Fまでの128文字しか扱うことができません。したがって&H80以上の文字、例えばカタカナを扱うことができません。このような場合に&H80以上の文字を扱うための方法がSI/SO制御です。SI/SO制御を有効にすると、データビットが7ビットでも&H00から&HFFまでのデータ通信ができるようになります。

SI/SO制御を指定するとバイナリファイル(データとして&H0E、&H0Fを含む可能性のあるファイル)の送受信を行うことはできません。これは&H0E、&H0Fが制御コードとして使用されているためです。

XON/XOFF 制御

データの転送速度がデータの処理速度より速い場合、受信バッファのあふれ(オーバーフロー)が発生する可能性があります。このような受信バッファのオーバーフローによるエラー(Line buffer overflow)を回避するためにバッファ制御機能が備わっています。

XON/XOFF 通信制御を指定するとバイナリファイル(データとして &H13、&H11 を含む可能性のあるファイル)の送受信を行うことはできません。これは &H13、&H11 が制御コードとして使用されているためです。

受信バッファの大きさは256バイトです。

8.2 データ通信

8.2.1 通信回線のオープン

通信回線のオープンは OPEN 文で行います。

(1) 形式

OPEN "COM1:[(通信パラメータ)]" FOR [INPUT | OUTPUT] AS #ファイル番号

ひとつの通信回線に対して入力モードと出力モードでオープンする場合は INPUT、OUTPUT を省略します。

通信パラメータの設定は「8.1 通信パラメータ」を参照してください。

(2) 制御信号線

信号線	動　作
DTR	オープンするとONになります。クローズするとOFFになります。
RTS	オープンするとONになります。クローズするとOFFになります。

8.2.2 データの出力

ファイルの書き出しと同じ命令を使って行います。

(1) 形式

PRINT#	データを送信します。
PRINT# USING	書式指定に従ってデータを送信します。
WRITE#	データを出力します。データとデータの間にカンマ(")を送信し、文字データは二重引用符(")で囲んで送信します。

(2) 制御信号線

信号線	動　作
CTS	CTS が ON になるまでデータを送信することはできません。

8.2.3 データの入力

ファイルからの読み込みと同じ命令を使って行います。

(1) 形式

INPUT#	データを受信します。
LINE INPUT#	CR コード(&H0D)までを1つのデータとし受信します。
INPUT$	指定した文字数分のデータを受信します。
EOF	受信バッファにデータがあるかないかを調べます。
LOC	受信バッファにあるデータ数を調べる。
LOF	受信バッファの空いている大きさを調べます。

(2) データ入力時のエラー

(Line buffer overflow)

受信バッファがオーバーフローしました。

8.2.4 通信回線のクローズ

CLOSE 文を用います。

CLOSE	ファイルをクローズします。

第9章

機械語モニタ

BASIC は BASIC のプログラムを実行する機能のほかに、直接機械語プログラムを作成したり、メモリやフロッピーディスクの内容を確認できるような機械語モニタと呼ばれる機能を持っています。

9.1　機械語モニタの起動と終了

9.1.1　メモリスイッチの設定

機械語モニタの機能の拡張部分はメモリスイッチ SW6-3 を 1 にすると、BASIC 起動時にシステムディスクからメモリに読み込まれます。したがって機械語モニタのすべての機能を使う場合はメモリスイッチ SW6-3 を 1 にして、BASIC を起動してください。BMENU では「メモリスイッチ変更」の中で「モニタモード」を「使用」にすることで変更できます。システム設定値は 0 ですので、通常は機械語モニタの一部の機能しか使用できません。

9.1.2　機械語モニタの起動と終了

機械語モニタは MON コマンドから起動します。

```
OK
MON [↵]
h]
```

機械語モニタに入ると、プロンプトとして"h］"を表示し、コマンド入力待ちになります。

機械語モニタの終了ーBASIC レベルへの復帰

コマンド入力待ちの状態から [CTRL] キーを押した下げたまま [B] キーを押します。

```
h] [CTRL]+[B]
Ok
```

9.1.3　機械語モニタの使用領域

機械語モニタを使うと直接メモリの内容を読み出したり、書き換えたりすることができます。特にメモリの内容が書き換わるようなコマンドを実行する場合には、次のようにして BASIC で使用する領域とは重ならない独立したメモリ領域を確保してください。

機械語モニタで機械語プログラムを作成したり、ディスクからデータを読み込む場合にもメモリの内容は書き換わりますので注意してください。

CLEAR ,&hXXXX [↵] ……………… CLEAR 文で指定したセグメントアドレス以降はBASICでは使用しない領域になります。

DEF SEG=&hXXXX [↵] ……………… 機械語モニタのセグメントアドレスを設定します。機械語モニタでセグメントアドレスを指定しない場合は、ここで設定したセグメントアドレスを指定したことになります。

MON [↵] ……………………………………… 機械語モニタの実行

h] CXXXX [↵] ……………………………… MON コマンドを実行する前に DEF SEG 文を実行しない場合は機械語モニタの中でCコマンドを実行します。

9.2 機械語モニタのコマンド

9.2.1 機械語モニタのコマンド

機械語モニタは次の19種類のコマンドを持っています。コマンドは英字1文字あるいは CTRL キーと組み合わせた1文字のコマンド名とパラメータから構成されます。パラメータにはメモリのアドレス、ディスクのトラック番号などを16進数で指定します。

*がついているコマンド(A, E, L, HELP, CTRL + D, CTRL + R, CTRL + W)はメモリスイッチ SW6-3が1の場合のみ使用可能です。

コマンド	機能	書式
A*	i8086のニーモニックをアセンブルしてメモリに格納	A [セグメントアドレス:] [オフセットアドレス]
C	セグメントアドレスの変更	C [セグメントアドレス]
D	メモリの内容を16進数で表示	D [開始アドレス] [, 終了アドレス]
E*	メモリの内容を表示して変更	E [セグメントアドレス:] [オフセットアドレス]
F	メモリの内容を指定した値で置き換える	F開始アドレス, 終了アドレス, 定数
G	機械語プログラムの実行	G [開始アドレス] [, 停止アドレス1] [, 停止アドレス2]
I	I/Oポートからデータ入力	Iポートアドレス
L*	メモリの内容を逆アセンブルして表示	L [セグメントアドレス:] [開始アドレス] [, 終了アドレス]
M	メモリ間のデータ転送	M転送元開始アドレス, 転送元終了アドレス, 転送先アドレス
O	I/Oポートへのデータ出力	Oポートアドレス
P	プリンタの出力制御	P
S	メモリとメモリスイッチの内容を変更	S [アドレス] SSW [メモリスイッチ]
X	CPUのレジスタやフラグの値を表示し、変更	X [レジスタ名] X [フラグ名]
HELP*	コマンドの書式を画面に表示 CTRL + A でも同じ機能になります	^A
CTRL + B	機械語モニタの終了	^B

コマンド	機能	書式
CTRL * + D	ディスクの内容をセクタ単位で表示	^Dドライブ番号，サーフェイス番号，トラック番号，開始セクタ番号［終了セクタ番号］
CTRL + P	コマンドの実行結果をプリンタに出力	^P
CTRL * + R	ディスクの内容をセクタ単位で読み込み	^Rドライブ番号，サーフェイス番号，トラック番号，セクタ番号，開始アドレス，終了アドレス
CTRL * + W	ディスクのセクタ単位でデータの書き込み	^Wドライブ番号，サーフェイス番号，トラック番号，セクタ番号，開始アドレス，終了アドレス

注意 書式のなかで角かっこ［ ］で囲まれたパラメータは任意に指定できる項目です。場合によっては省略することができます。

コマンド名

コマンド名は大文字で入力しても、小文字で入力しても構いません。CTRL + A は CTRL キーを押し下げたまま A キーを押すことを意味します。画面上には^Aと表示されます。

アドレス

コマンドにはセグメントアドレス：オフセットアドレスの形式でセグメントアドレスを指定できるものと、オフセットアドレスしか指定できないものがあります。アドレスはどちらも4桁までの16進数です。

セグメントアドレスを省略した場合、あるいはセグメントアドレスを指定できないコマンドではセグメントアドレスとしてCコマンドで指定したアドレスが使用されます。
オフセットアドレスを省略した場合は直前に指定したコマンドの終了アドレスの次のアドレスを指定したことになります。

機械語モニタ起動時のセグメントアドレスはDEF SEG文で指定したアドレスになります。またBASIC起動時のセグメントアドレスは&hA000、オフセットアドレスは&h0000になります。

ポートアドレス

ポートアドレスを指定しするための4桁までの16進数です。

ドライブ番号，サーフェイス番号，トラック番号，セクタ番号

ディスクのセクタの位置を16進数で指定します。指定できる範囲はディスクの種類によって異なりますので「第13章　ディスクのファイル管理」を参照してください。1MBフロッピーディスクの0トラックは1セクタ128バイトでフォーマットされています。したがって読み出したり書き出したりする場合は128バイト単位になります。その他のセクタは256バイトです。

9.2.2 コマンドの詳細

A Assemble(アセンブル)

機能 i8086のニーモニックを1行単位にアセンブルして、メモリに格納します。

書式 A [セグメントアドレス:] [オフセットアドレス]

説明
- アドレスを省略した場合は、直前に実行したAまたはLコマンドの実行終了アドレスから格納します。これらのコマンドを実行していない場合はオフセットアドレスは0になります。
- アセンブルを終了するには、次のいずれかを実行します。
  アドレスを表示しているときに、[↵]キーだけ押します。
  [STOP]キーまたは[CTRL]+[C]キーを押します。
- 現在のセグメントアドレスで指定できないアドレスに制御を移す命令(例えば、CALL, JMP, RETなど)にはニーモニックにFを付けて入力します。

  例 | JMPF SSSS:OOOO　　SSSS:セグメントアドレス
  　　　　　　　　　　　OOOO:オフセットアドレス
- 入力した行に文法的な誤りがある場合は、次の行の11桁目から???を表示し、同じアドレスで再度入力待ちになります。

  例
  ```
  ADDF AX, 100
  ???
  ```
- オペランドの数値に四則演算の結果を指定することができます。

  例 | MOV AX, 1 * 3 * 4
- リピートプリフィクスとセグメントオーバライドプリフィクスは同一行に記述できますが、セグメントオーバライドプリフィックから機械語に変換します。
- 数値はすべて16進数で入力します。文字コードを入力する場合には'C'のように引用符で囲みます。
- アドレスにイミディエイト・オフセットを指定する場合は必ず角かっこ([)の前に記述します。
- 現在のプログラムカウンタを'$'で参照できます。
- 疑似命令としてDB(1バイト)、DW(2バイト)を使用できます。

  例
  ```
  DB  1, 2, 3
  DB  3, 'ABC', 0D, 0A
  DW  0, 8000, FFFF
  ```

```
h]c8000 …セグメントアドレスを&H8000に設定します
h]a0 …0番地からアセンブルします
8000:0000 50              PUSH AX
8000:0001 53              PUSH BX
8000:0002 51              PUSH CX
8000:0003 52              PUSH DX
8000:0004 56              PUSH SI
8000:0005 1E              PUSH DS
8000:0006 06              PUSH ES
8000:0007 muv ax,word 06[bx]…入力した行に誤りがあると"???"を次の行に表示します
          ???
8000:0007 8B4706          MOV AX,WORD 06[BX]
8000:000A 8EC1            MOV ES,CX
8000:000C 8B7704          MOV SI,WORD 04[BX]
8000:000F 26:8B34         MOV SI,WORD ES:[SI]
8000:0012
```

C

機能 セグメントアドレスを設定します。

書式 C[セグメントアドレス]

解説
- セグメントアドレスを省略すると現在のセグメントアドレスを表示します。
- 機械語モニタ起動時のセグメントアドレスはDEF SEG文で指定したアドレスになります。DEF SEG文を実行してない場合はBASICの上限(VRAMテキスト領域&hA000)になります。

```
h]c  …現在のセグメントアドレスを調べます
A000
h]c8000…セグメントアドレスを&H8000に変更します
h]c
8000
h]
```

D　　Dump(ダンプ)

機能 指定範囲のメモリの内容を画面に16進数と文字で表示します。

書式 D[開始アドレス][,終了アドレス]

解説
- 開始アドレスから終了アドレスまでの内容を16進数表記と対応する文字で表示します。
- 制御コードなど、対応する文字を表示できない場合はピリオド(.)を表示します。
- 開始アドレスを省略した場合は、その前に実行したDコマンドの最終アドレスの次のアドレスを指定したことになります。前にDコマンドを実行してなければ、0を指定したことになります。
- 終了アドレスを省略した場合は、開始アドレスから16バイト分の内容を表示します。
- 表示を中止するには [STOP] キーか [CTRL]＋[C] キーを押します。
- 表示を一時停止するには [CTRL]＋[S] キーを押します。再開するには [STOP] キーまたは [CTRL]＋[C] キー以外のキーを押します。
- Dコマンド終了後に [↵] キーを入力すると次のアドレスから16バイトの内容を表示します。
- オフセットアドレス0から&HFFFFまでの範囲の内容を表示することができます。

```
h]c0 …セグメントアドレスを0に変更します
h]d0,ff …システムエリアの内容を調べます
0000 3A 01 00 F9 36 07 80 FD 2C 06 80 FD 36 07 80 FD   :   6 _ , _ 6 _
0010 36 07 80 FD A2 00 20 F3 76 06 20 F3 36 07 80 FD   6 _ ｢  月v  月6 _
0020 21 19 80 FD 80 0A 80 FD 6C 06 80 FD 6C 06 80 FD   ! _ _ _ l _ l _
0030 20 14 80 FD C7 01 00 F9 6C 06 80 FD 6C 06 80 FD     _ ｽ   l _ l _
0040 58 06 80 FD 65 0F 00 D7 7F 1C 80 FD BB 1B 80 FD   X _ e  ﾗ   _ ｻ _
0050 6C 06 80 FD 6C 06 80 FD 58 06 80 FD 6C 06 80 FD   l _ l _ X _ l _
0060 B5 5F 00 16 25 15 80 FD 6E 26 80 FD A8 19 80 FD   ｵ_  % _ n&_ ｨ _
0070 3E 18 80 FD 36 07 80 FD F7 25 00 E8 06 00 00 F9   > _ 6 _ 秒% ♠
0080 A6 03 00 E8 A6 03 00 E8 A6 03 00 E8 A6 03 00 E8   ｦ  ♠ｦ  ♠ｦ  ♠ｦ  ♠
0090 A6 03 00 E8 A6 03 00 E8 A6 03 00 E8 A6 03 00 E8   ｦ  ♠ｦ  ♠ｦ  ♠ｦ  ♠
00A0 A6 03 00 E8 A6 03 00 E8 A6 03 00 E8 A6 03 00 E8   ｦ  ♠ｦ  ♠ｦ  ♠ｦ  ♠
00B0 A6 03 00 E8 A6 03 00 E8 A6 03 00 E8 A6 03 00 E8   ｦ  ♠ｦ  ♠ｦ  ♠ｦ  ♠
00C0 A6 03 00 E8 A6 03 00 E8 A6 03 00 E8 A6 03 00 E8   ｦ  ♠ｦ  ♠ｦ  ♠ｦ  ♠
00D0 A6 03 00 E8 A6 03 00 E8 A6 03 00 E8 A6 03 00 E8   ｦ  ♠ｦ  ♠ｦ  ♠ｦ  ♠
00E0 A6 03 00 E8 A6 03 00 E8 A6 03 00 E8 A6 03 00 E8   ｦ  ♠ｦ  ♠ｦ  ♠ｦ  ♠
00F0 A6 03 00 E8 A6 03 00 E8 A6 03 00 E8 A6 03 00 E8   ｦ  ♠ｦ  ♠ｦ  ♠ｦ  ♠
h]… [↵] キーだけ押すと次の16バイトの内容を表示します
```

E　Edit memory(エディット・メモリ)

機能　メモリの内容を画面に表示し、スクリーンエディタを用いてメモリの内容を変更します。

書式　E[アドレス]

解説
- 指定したアドレスから16行分のメモリの内容を表示します。
- 画面上でカーソルを移動し数値を入力すると、表示内容とそれに対応するメモリの内容を修正します。
- カーソルはカーソル移動キー [→]、[←]、[↑]、[↓] キーで自由に移動することができます。
- [ROLL UP] キー、[ROLL DOWN] キーを押すと表示しているアドレスの内容を1行ずつスクロールすることができます。
- アドレスを省略すると、直前に実行したEコマンドの実行終了時の画面状態を再現します。カーソルも直前のアドレスに移動します。Eコマンドを実行していなければ0を指定したことになります。
- Eコマンドを終了するには次のキーを押します。

  [STOP] キーまたは [CTRL] + [C] キー
  [ESC] キー
- Eコマンドを実行する場合はスクロール範囲を17行以上に設定してください。17行以下の設定だと"?"を表示しEコマンドを実行できません。
- Eコマンドを実行する場合に画面が40桁モードになっていると、Eコマンド実行時のみ80桁モードに変更して画面を表示します。
- オフセットアドレスが&HFFFFを越える場合は、同一セグメントの&H0000に続きます。

```
h]c8000 …セグメントアドレスを&H8000に設定します
h]e     …エディットモードに入ります
```

```
8000:0000 ;50 53 51 52 56 1E 06 8B-4F 06 8E C1 8B 77 04 26 PSQRV..▮O.▮ﾁ▮w.&
8000:0010 ;8B 04 C5 1F 8B 0F 8B 5F-02 08 ED 74 03 BA 60 00 ▮.ﾅ.▮.▮_..Ot.ｺ'.
8000:0020 ;8E DA B9 10 00 31 F6 30-F6 D1 C0 D0 D6 80 C6 30 ▮ﾚｹ..1♀0♀ﾑﾀﾐﾖ_ﾆ0
8000:0030 ;88 30 46 E2 F2 07 1F 5E-5A 59 5B 58 CF 00 00 00 |0F≠年..^ZY[Xﾏ...
8000:0040 ;00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ................
8000:0050 ;00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ................
8000:0060 ;00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ................
8000:0070 ;00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ................
8000:0080 ;00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ................
8000:0090 ;00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ................
8000:00A0 ;00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ................
8000:00B0 ;00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ................
8000:00C0 ;00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ................
8000:00D0 ;00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ................
8000:00E0 ;00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ................
8000:00F0 ;00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ................
→ ← ↑ ↓ (MOVE CURSOR)   ROLL UP/ROLL DOWN (SCROLL SCREEN)   STOP/ESC (QUIT)
```

F　Fill(フィル・メモリ)

機能　指定した範囲のメモリの内容を指定した値に書き換えます。

書式　F 開始アドレス, 終了アドレス, 定数

解説
- 開始アドレスから終了アドレスまでの内容をすべて定数の値に書き換えます。
- オフセットアドレス0から&HFFFFまでの範囲の内容を変更することができます。

```
h]d100,11f
0100 B8 00 01 BB 23 01 01 D8 4F 06 8E C1 8B 77 04 26    ｸ  ｻ#  ﾘO ■ｹ■w &
0110 8B 04 C5 1F 8B 0F 8B 5F 02 08 ED 74 03 BA 60 00    ■ ﾅ ■ ■_  Ot ｺ'
h]f100,11f,0
h]d100,11f
0100 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0110 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
h]
```

G　　　　Go(ゴー)

機能 指定したアドレスからプログラムを実行します。

書式 G［開始アドレス］［，停止アドレス１］［，停止アドレス２］

解説
- 開始アドレスからプログラムを実行します。
- 停止アドレス(ブレイクポイントアドレス)を２つ設定することができます。
- 開始アドレスを省略すると、直前に実行したＧコマンドの停止アドレスからプログラムを実行します。
- 停止アドレスでプログラムの実行が停止した場合は、レジスタの内容は保存されています。
- 停止アドレスに達すると、そのアドレスを表示してプログラムの実行を停止します。

```
h]a100 …&H100番地からアセンブルします
8000:0100 B80001         MOV AX,100
8000:0103 BB2301         MOV BX,123
8000:0106 01D8           ADD AX,BX
8000:0108
h]g100,108…&H107番地まで実行します
*8000:0108
h]xax
AX-0223 = …結果を調べると&H223になっています
```

I　　　　Input(インプット)

機能 指定したI/Oポートからデータを読み込み、画面に表示します。

書式 Ｉポートアドレス

解説
- I/Oポートからデータを読み込み、次の行に表示します。
- ポートアドレスは16進数で０から&hFFFFの範囲です。
- データ表示後、次のデータ入力待ちになります。この状態で［↵］キーを押すと終了します。［スペース］キーを押すと次のデータを表示して、再び入力待ちになります。

L　　　　List(ここでは逆アセンブルの意味)

機能 指定範囲のメモリの内容を逆アセンブルし、i8086のニーモニックで表示します。

書式 Ｌ［セグメントアドレス：］開始アドレス［，終了アドレス］

解説
- 開始アドレスから終了アドレスまでのメモリの内容を逆アセンブルします。
- 終了アドレスを省略した場合は、開始アドレスからテキスト画面の有効範囲の約半分が埋まる分のメモリの内容を表示します。

・開始アドレスを省略すると、直前に実行したAまたはLコマンドの実行終了アドレスの次のアドレスを指定したことになります。
・実行を終了するには STOP キーまたは CTRL ＋ C キーを押します。
・実行を一時停止するには CTRL ＋ S キーを押します。再開するには STOP キーまたは CTRL ＋ C キー以外のキーを押します。
・機械語命令として認識できないコードを発見したときは、ニーモニックの表示部分に"???"を表示して、次のアドレスから処理を続けます。
・オフセットアドレス0から&HFFFF までの範囲の内容を逆アセンブルすることができます。

```
h]l0 …0番地から逆アセンブルします
8000:0000 50              PUSH    AX
8000:0001 53              PUSH    BX
8000:0002 51              PUSH    CX
8000:0003 52              PUSH    DX
8000:0004 56              PUSH    SI
8000:0005 1E              PUSH    DS
8000:0006 06              PUSH    ES
8000:0007 8B4F06          MOV     CX,WORD 06[BX]
8000:000A 8EC1            MOV     ES,CX
8000:000C 8B7704          MOV     SI,WORD 04[BX]
8000:000F 26              ES:
h]
```

M　　Move(ムーブメモリ)

機能 指定範囲のメモリの内容を指定アドレスから始まる領域に転送します。

書式 M 転送元開始アドレス, 終了アドレス, 転送先アドレス

解説 ・転送元開始アドレスから終了アドレスの領域の内容を、そのままの順番で、転送先アドレスから始まる領域へ転送します。
・転送先アドレスが転送領域内にある場合も、転送することができます。

```
h]d0,1f …0番地から&H1F番地をダンプします
0000 50 53 51 52 56 1E 06 8B 4F 06 8E C1 8B 77 04 26      PSQRV ▮O ■ｹ▮w &
0010 8B 04 C5 1F 8B 0F 8B 5F 02 08 ED 74 03 BA 60 00      ▮ ﾅ ▮ ▮_  Ot ｺ'
h]d100,11f …&H100番地から&H11F番地をダンプします
0100 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0110 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
h]m0,1f,100…0番地から&H1F番地の内容を&H100番地からにコピーします
h]d100,11f …確かにコピーされています
0100 50 53 51 52 56 1E 06 8B 4F 06 8E C1 8B 77 04 26      PSQRV ▮O ■ｹ▮w &
0110 8B 04 C5 1F 8B 0F 8B 5F 02 08 ED 74 03 BA 60 00      ▮ ﾅ ▮ ▮_  Ot ｺ'
h]
```

O　　Output(アウトプット)

機能 指定したI/Oポートへデータを出力します

書式 O ポートアドレス, データ

解説 ・I/Oポートへデータを書き込みます。
・ポートアドレスは16進数で0から&HFFFF、データは0から&HFFの範囲になります。

P

Print(プリント)

[機能] D，L，[CTRL]＋[D]コマンドの実行結果をプリンタに出力します。

[書式] P

[解説] • Pコマンドは切換スイッチになっており、入力するたびに設定の状態が変わります。

• 現在どの設定になっているかはプロンプトの形で知ることができます。

h］　プリンタに出力しない状態

h）　プリンタに出力する状態

• 機械語モニタ起動時の設定はプリンタに出力しない状態です。

S

Set(セット)

[機能] (1)指定したアドレスの内容を表示し、そのデータを変更します。

(2)指定したメモリスイッチの内容を表示し、その内容を変更します。

[書式] (1) S[アドレス]　メモリの内容変更

(2) SSW[1－7]　メモリスイッチの内容変更

[解説] (1) S［アドレス］

• 指定したアドレスの内容を表示し、入力待ちの状態になります。メモリの内容を変更する場合は、16進数で0から&hFFのデータを入力します。データに続けて[スペース]キーを押せば、メモリの内容を変更して、次のアドレスの内容を表示します。[↵]キーを押せば、メモリの内容を変更し、コマンドの実行を終了します。

```
h]d100,11f …&H100番地から&H11F番地ダンプします
0100 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0110 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
h]s100 …&H100番地からメモリの内容を変更します
0100 00-20 00-21 00-22 00-23 00-24 00-25 00-26 00-27
0108 00-28 00-29 00-2a 00-2b 00-2c 00-2d 00-2e 00-2f
h]d100,11f
0100 20 21 22 23 24 25 26 27 28 29 2A 2B 2C 2D 2E 2F          !"#$%&'()*+,-./
0110 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
h]
```

• データを入力しないでスペースキーを押すと、内容を変更せずに、次のアドレスの内容を表示します。

(2) SSW［1－7］

• 数値に対応するメモリスイッチの内容を表示した後、入力待ちの状態になります。メモリスイッチの内容を変更する場合は、16進数でデータを入力します。データに続けて[スペース]キーを押せば、メモリスイッチの内容を変更して、次のメモリスイッチの内容を表示します。[↵]キーを押せば、メモリスイッチの内容を変更してコマンドの実行を終了します。

```
h]ssw …メモリスイッチの内容を調べます
SW1 SW2 SW3 SW4 SW5 SW6 SW7
48  05  04  00  01  08  00
h]ssw6 …メモリスイッチ SW6 の内容を変更します
08-18
h]ssw
SW1 SW2 SW3 SW4 SW5 SW6 SW7
48  05  04  00  01  18  00
h]
```

- データを入力しないでスペースキーを押すと、内容を変更せずに、次のメモリスイッチの内容を表示します。
- メモリスイッチの番号を省略すると、すべてのメモリスイッチの内容を表示します。

注意 メモリスイッチの内容はBASIC起動時に読み込まれます。このためメモリスイッチの内容を変更した後は必ず、BASICを再起動してください。再起動した時点からメモリスイッチの内容は有効になります。

X　　eXamine(イグザミン)

機能 CPUのレジスタやフラグの内容を表示し、変更します。

書式 X[レジスタ名]

X[フラグ名]

解説
- 指定したレジスタやフラグの内容を表示し、データの入力待ちになります。
- 指定できるレジスタとフラグ名は次のとおりです。

レジスタ名　CS, DS, SS, ES, SP, BP, SI, DI, AX, BX, CX, DX, IP

フラグ名　O(オーバーフロー), D(ディレクション), I(インターラプトイネーブル), T(トラップ), S(サイン), Z(ゼロ), A(補助キャリー), P(パリティ), C(キャリー)

- レジスタ名を指定してデータを入力した場合は、[↵]キーを押すと指定したレジスタの内容を変更し、コマンドの実行を終了します。[↵]キーの代わりに[スペース]キーを押すと、次のレジスタの内容を表示し、入力待ちになります。データを入力せずに[↵]キーまたは[スペース]キーを押すとレジスタの内容を表示しません。最後のレジスタ(IP)の処理が終わるとコマンドの実行を終了します。
- フラグ名を指定すると、そのフラグの現在の状態を表示し、データ入力待ちになります。データを入力し[↵]キーまたは[スペース]キーを押すと、フラグの状態を変更してコマンドの実行を終了します。
- 変更したレジスタやフラグの内容が有効になるのは、Gコマンドを実行するときだけです。
- 途中で[STOP]キーや[CTRL]+[C]キーを押すとコマンドの実行を終了します。そのときのデータ入力待ちのレジスタやフラグの内容は変更されません。

```
h]x …レジスタ，フラグの値を調べます
FL ---------
CS-8000    DS-0000    SS-0060    ES-0000
SP-F7F6    BP-0000    SI-0000    DI-0000
AX-0223    BX-0123    CX-0000    DX-0000
IP-0108
h]
```

HELP または CTRL + A

機能 コマンドの種類と使用方法を表示します。

書式 コマンド入力待ちで HELP キーまたは CTRL + A キーを押す

解説 機械語モニタで使用できるの種類とその書式を画面に表示します。

CTRL + B

機能 機械語モニタを終了しBASICに戻ります。

書式 コマンド入力待ちで CTRL + B キーを押す。

解説 機械語モニタを終了しBASICに戻ります。

CTRL + D

機能 ディスクの指定セクタの内容を表示します。

書式 CTRL + D ドライブ番号，サーフェス番号，トラック番号，開始セクタ番号 [,終了セクタ番号]

解説
- 指定したドライブ番号のディスクの指定サーフェス番号の、トラック番号上の開始セクタから終了セクタまでの内容をセクタ順に表示していきます。
- 複数のトラックにまたがって表示することはできません。
- 終了セクタ番号を省略すると、開始セクタ番号のセクタだけを表示します。
- 表示を途中で終了するには、STOP キーまたは CTRL + C キーを押します。
- 表示を一時停止するには、CTRL + S キーを押します。再開するには STOP キーまたは CTRL + C キー以外のキーを押します。

```
h]^D1,0,23,1  …ドライブ1, サーフェイスD, トラック35, セクタ1(ディレクトリ領域)をダンプします
Drive=01 Surface=00 Track=0023 Sector=01

0000 65 67 64 69 63 6D 64 69 63 00 47 FF FF FF FF FF        egdicmdic G
0010 65 67 64 69 63 75 64 69 63 00 45 FF FF FF FF FF        egdicudic E
0020 65 67 7A 69 70 6D 64 69 63 00 41 FF FF FF FF FF        egzipmdic A
0030 65 67 7A 69 70 75 64 69 63 00 38 FF FF FF FF FF        egzipudic 8
0040 42 4D 45 4E 55 20 20 20 20 80 37 FF FF FF FF FF        BMENU     _7
0050 62 6D 65 6E 75 20 20 20 20 80 33 FF FF FF FF FF        bmenu     _3
0060 42 4D 45 4E 55 20 41 55 54 80 32 FF FF FF FF FF        BMENU AUT_2
0070 42 4D 45 4E 55 20 42 4B 50 80 30 FF FF FF FF FF        BMENU BKP_0
0080 62 6D 65 6E 75 20 62 6B 70 01 2F FF FF FF FF FF        bmenu bkp /
0090 42 4D 45 4E 55 20 43 50 59 80 2E FF FF FF FF FF        BMENU CPY_.
00A0 62 6D 65 6E 75 20 63 70 79 01 2C FF FF FF FF FF        bmenu cpy ,
00B0 42 4D 45 4E 55 20 44 43 4D 80 2B FF FF FF FF FF        BMENU DCM_+
00C0 42 4D 45 4E 55 20 44 43 31 80 29 FF FF FF FF FF        BMENU DC1_)
00D0 42 4D 45 4E 55 20 44 43 32 80 28 FF FF FF FF FF        BMENU DC2_(
00E0 62 6D 65 6E 75 20 64 63 64 01 27 FF FF FF FF FF        bmenu dcd '
00F0 62 6D 65 6E 75 20 64 63 6A 01 1F FF FF FF FF FF        bmenu dcj
h]
```

CTRL + R

機能 ディスク上の指定セクタの内容をメモリに読み込みます。

書式 CTRL + R ドライブ番号，サーフェス番号，トラック番号，セクタ番号，開始アドレス，終了アドレス

解説 • ドライブ番号，サーフェス番号，トラック番号，セクタ番号で指定したセクタの内容をメモリ上の開始アドレスから終了アドレスまでのバイト数だけ読み込みます。

```
h]c9000…ドライブ1, サーフェイス0, トラック35, セクタ1からセクタ22(ディレクトリ領域)を0番地から読み込みます
h]^R1,0,23,1,0,15ff
h]d0,ff
0000 65 67 64 69 63 6D 64 69 63 00 47 FF FF FF FF FF        egdicmdic G
0010 65 67 64 69 63 75 64 69 63 00 45 FF FF FF FF FF        egdicudic E
0020 65 67 7A 69 70 6D 64 69 63 00 41 FF FF FF FF FF        egzipmdic A
0030 65 67 7A 69 70 75 64 69 63 00 38 FF FF FF FF FF        egzipudic 8
0040 42 4D 45 4E 55 20 20 20 20 80 37 FF FF FF FF FF        BMENU     _7
0050 62 6D 65 6E 75 20 20 20 20 80 33 FF FF FF FF FF        bmenu     _3
0060 42 4D 45 4E 55 20 41 55 54 80 32 FF FF FF FF FF        BMENU AUT_2
0070 42 4D 45 4E 55 20 42 4B 50 80 30 FF FF FF FF FF        BMENU BKP_0
0080 62 6D 65 6E 75 20 62 6B 70 01 2F FF FF FF FF FF        bmenu bkp /
0090 42 4D 45 4E 55 20 43 50 59 80 2E FF FF FF FF FF        BMENU CPY_.
00A0 62 6D 65 6E 75 20 63 70 79 01 2C FF FF FF FF FF        bmenu cpy ,
00B0 42 4D 45 4E 55 20 44 43 4D 80 2B FF FF FF FF FF        BMENU DCM_+
00C0 42 4D 45 4E 55 20 44 43 31 80 29 FF FF FF FF FF        BMENU DC1_)
00D0 42 4D 45 4E 55 20 44 43 32 80 28 FF FF FF FF FF        BMENU DC2_(
00E0 62 6D 65 6E 75 20 64 63 64 01 27 FF FF FF FF FF        bmenu dcd '
00F0 62 6D 65 6E 75 20 64 63 6A 01 1F FF FF FF FF FF        bmenu dcj
h]
```

CTRL + W

機能 ディスク上の指定セクタにメモリの内容を書き込みます。

書式 CTRL + W ドライブ番号，サーフェス番号，トラック番号，セクタ番号，開始アドレス，終了アドレス

解説 • ドライブ番号，サーフェス番号，トラック番号，セクタ番号で指定したセクタに、メモリ上の開始アドレスから終了アドレスまでの内容をセクタの先頭から詰めて書き出します。

CTRL + P

機能 プリンタの出力制御を行います。

書式 CTRL + P

解説 • 次のコマンドの実行結果をプリンタに出力します。

A, E, HELP (CTRL + A), SSW, X, G

• CTRL + P コマンドは切換スイッチになっており、入力するたびに設定の状態が変わります。

• 現在の状態はプロンプトの形で知ることができます。

h]　プリンタに出力しない状態

h/　プリンタに出力する状態

• PコマンドではD, L, CTRL + D コマンドの実行結果をプリンタに出力します

```
h]^P …プリンタに出力する設定にします
h/^P …プリンタに出力しない設定にします
h]
```

第10章

機械語プログラム

本章ではBASICプログラムから機械語プログラムを呼び出して実行する方法を説明します。ここで取り上げる項目は次のとおりです。

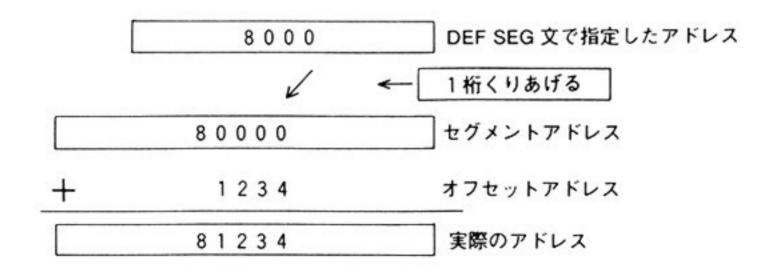
(1)アドレスの指定
　機械語プログラムをメモリのどこに格納すればよいのか、またそのときのアドレス指定はどうするのか
(2)機械語プログラムの格納方法
　機械語プログラムをメモリに格納するにはどうするのか
(3)機械語プログラムの実行
　BASICから機械語プログラムを実行するにはどうするのか、またBASICと機械語のデータ(引数)の受け渡し方法はどうするのか

なお説明は、ある程度機械語のプログラミングを理解できる方を対象にして行います。

10.1　アドレスの指定

BASICは1MBのメモリ空間をセグメントアドレスとオフセットアドレスを組み合わせて指定します。セグメントアドレスを指定するのがDEF SEG文で、DEF SEG文以外のPEEK, POKE文などのアドレスを指定する命令はすべてオフセットアドレスを指定することになります。

セグメントアドレスには16進数で0から&HFFFFまでの数値を指定します。この値を16進数で1桁上げ(最後の桁に0を付ける)、みかけ上20ビットにします。これにオフセットアドレスを加えたものが実際のメモリアドレスになります。オフセットアドレスにも16進数で0から&HFFFFまでの数値を指定します。図にセグメントアドレスとオフセットアドレスの関係を示します。

例えばセグメントアドレスが&H8000、オフセットアドレスが&H1234であれば、&H80000＋&H1234で&H81234番地を指定したことになります。

実際のアドレスを指定する方法は1つとは限りません。例えばアドレス&H12345を指定する場合、セグメントアドレス&H1000、オフセットアドレス&H2345と考えることもできますし、セグメントアドレス&H1230、オフセットアドレス&H45とも考えることもできます。

10.2 機械語プログラムの格納方法

機械語プログラムはBASICの持っていない命令や、BASICでは実行するのに時間のかかるプログラムを直接、機械語プログラムとして実行する場合などに使用します。

例 マウスドライバ
BASICはマウスを直接制御する命令は持っていません。しかし機械語プログラムのユーティリティとして用意していますので、これを実行すると簡単にマウスを使用することができます。
詳細については「第12章　マウス」を参照してください。

機械語プログラムはBASICの外で実行するため、もしエラーがあったとしてもBASICのプログラムのようにエラーメッセージを表示したりしません。機械語プログラムが暴走して、コンピュータを最初から立ち上げ直さなければならないこともあります。

機械語プログラムはCPUの持っている命令そのものを使用するためCPUのことをよく知らなければ作ることができません。

簡単な(小さい)プログラムであれば機械語モニタを使って作成することができます。

10.2.1 機械語プログラム領域の確保

機械語モニタを使って機械語プログラムを作成したり、機械語プログラムを実行する場合は、BASICのプログラムやデータを壊すことのないように、またBASICのプログラムやデータにより機械語プログラムが壊されないように、BASICで使用する領域とは重ならない独立したメモリ領域を確保する必要があります。

この設定をするのがCLEAR文です。CLEAR文で指定したセグメントアドレスからメモリの上限までを機械語プログラム領域として使用します。

CLEAR, プログラム領域, [スタック領域], [配列変数領域]

10.2.2 機械語プログラムの用意

機械語プログラムをメモリに格納するのには、次の2つの方法があります。

- 短い機械語プログラムであれば、POKE文を使い1バイトずつメモリに書き込みます。
- BLOADコマンドを使って、機械語プログラムをファイルとしてメモリに読み込みます。

機械語モニタでのプログラムの作成

Aコマンドで作成した機械語プログラムを次のようにしてディスクに保存すればBLOAD文でいつでも呼び出すことができます。

① 機械語モニタを起動します。

MON ⏎

② BASIC起動時にはセグメントアドレスをテキスト用VRAMの先頭(&HA0000番地)に設定します。このため、このまま機械語プログラムを書き込んで行くと画面上に意味のない文字を表示します。したがって、セグメントアドレスをCLEAR文で設定した機械語プログラム用のアドレスに変更します。また機械語モニタを起動する前にDEF SEG文でセグメントアドレスを変更することもできます。

h] C8000 ⏎

(3) 機械語プログラムを入力していきます。このプログラムは整数を2進数表記の文字列に変換するものです。

```
CLEAR ,&H8000
OK
DEF SEG=&H8000
OK
MON
h]A
8000:0000 50          PUSH AX
8000:0001 53          PUSH BX
8000:0002 51          PUSH CX
8000:0003 52          PUSH DX
8000:0004 56          PUSH SI
8000:0005 1E          PUSH DS
8000:0006 06          PUSH ES
8000:0007 8B4F06      MOV CX,WORD 06[BX]
8000:000A 8EC1        MOV ES,CX
8000:000C 8B7704      MOV SI,WORD 04[BX]
8000:000F 26:8B04     MOV AX,WORD ES:[SI]
8000:0012 C51F        LDS BX,WORD [BX]
8000:0014 8B0F        MOV CX,WORD [BX]
8000:0016 8B5F02      MOV BX,WORD 02[BX]
8000:0019 08ED        OR CH,CH
8000:001B 7403        JE 20
8000:001D BA6000      MOV DX,0060
8000:0020 8EDA        MOV DS,DX
8000:0022 B91000      MOV CX,0010
8000:0025 31F6        XOR SI,SI
8000:0027 30F6        XOR DH,DH
8000:0029 D1C0        ROL AX,01
8000:002B D0D6        RCL DH,01
8000:002D 80C630      ADD DH,30
8000:0030 8830        MOV BYTE [BX+SI],DH
8000:0032 46          INC SI
8000:0033 E2F2        LOOP 0027
8000:0035 07          POP ES
8000:0036 1F          POP DS
8000:0037 5E          POP SI
8000:0038 5A          POP DX
8000:0039 59          POP CX
8000:003A 5B          POP BX
8000:003B 58          POP AX
8000:003C CF          IRET
8000:003D
h]
```

④ 機械語プログラムを入力し終わったら、実行する前にまずセーブしてください。もし正常に実行せず、暴走してしまった場合などに、最初から入力し直す手間を省くことができます。

h] ^B…………………… [CTRL]＋[B]キーを押します。

DEF SEG &h8000

BSAVE "BIN . COM", 0 , 255

⑤ 機械語モニタからBASICのモードに戻っても、作成したプログラムはメモリ中に残っています。BASICプログラムから、この機械語プログラムを呼び出して実行します。

短いプログラムであればDATA文でデータとして持ち、これをREAD文で読み出しながらPOKE文でメモリに書き込む方法もあります。

10.3 機械語プログラムの実行

読み込んだ機械語プログラムはCALL文あるいはUSR関数により実行します。CALL文とUSR関数では機械語プログラムに渡すパラメータの設定方法が異なります。

10.3.1 CALL文

CALL文の書式は次のとおりです。

CALL 変数名［(引数1 ［, 引数2・・・］)］

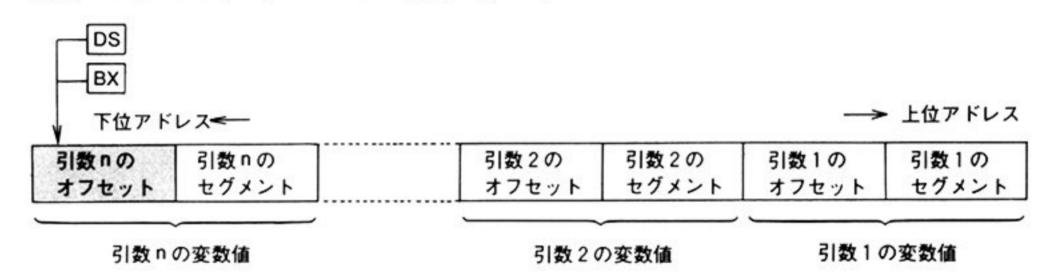
変数名は機械語プログラムセグメント内の実行開始アドレスを持つ変数の名前です。引数はBASICプログラムから機械語プログラムに渡すデータを持った変数の名前です。

引数がある場合は、それらの変数のアドレスを持った引数テーブルのアドレスがレジスタDSおよびBXに渡されます。

DS：引数テーブルのセグメントアドレス

BX：引数テーブルのオフセットアドレス

各引数に対応するデータの格納アドレスは次式で与えられます。

引数nのオフセットアドレス＝4×(引数の数－n)

(引数 n のオフセット)には引数nのデータが格納されている領域のオフセットアドレスが、(引数 n のセグメント)には引数 n のデータが格納されている領域のセグメントアドレスが格納されています。ここで示すアドレスはそれぞれ次の場所を示します。これは VARPTR 関数で示すアドレスと同じ場所になります。

	下位アドレス←――　　　　――→上位アドレス
整数型	□□（2バイト）
単精度型	□□□□（4バイト）
倍精度型	□□□□□□□□（8バイト）
文字型	□□□□（4バイト：ストリングディスクリプタ）

□ 引数テーブルの示すアドレス

それぞれのデータの形式は「第11章　データの内部構造」を参照してください。

機械語プログラムは引数テーブルに設定されたアドレスを参照して BASIC にデータを返すことができます。

機械語プログラムから BASIC プログラムに戻るには、必ず IRET 命令を使ってください。

プログラム例

「機械語プログラムの作成」で作成した機械語プログラムを BASIC プログラムから実行します。機械語プログラムは BIN.COM というファイル名で BSAVE コマンドでセーブしておいてください。

INPUT 文で整数を与え、その整数を 2 進数表記の文字列に変換します。

```
100 '   整数を2進数に変換
110 '
120 CLEAR ,&H8000          ' 機械語プログラム領域の確保
130 DEF SEG=&H8000         ' 機械語プログラムセグメント
140 BLOAD "2:BIN.COM"
150 '
160 BIN=0                  ' 機械語プログラム実行開始アドレス
170 B$=SPACE$(16)
180 '
190 INPUT A%
200     CALL BIN(A%,B$)
210     PRINT A%,B$
220 GOTO 190
230 '

RUN
? 25
 25           0000000000011001
? 6523
 6523         0001100101111011
? -529
-529          1111110111101111
?
```

10.3.2 USR関数

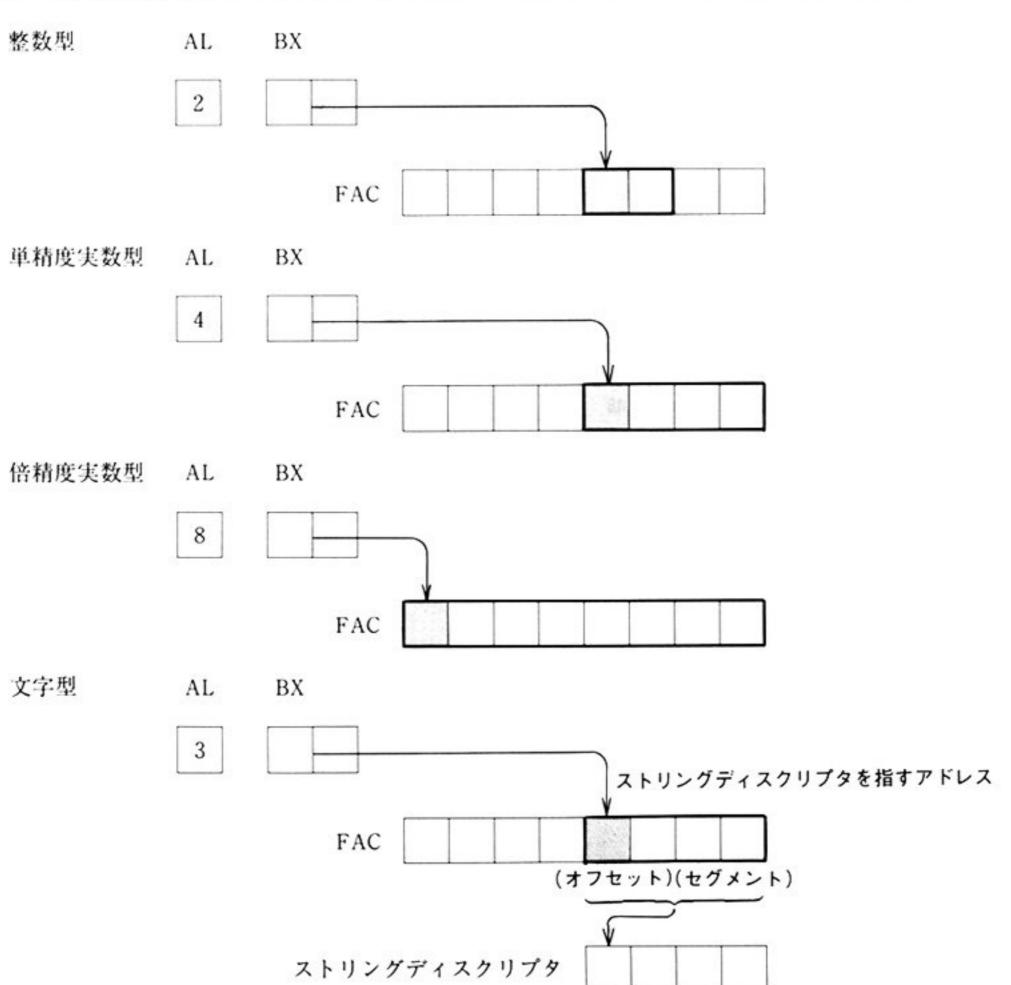
USR関数の書式は次のとおりです。

USR〔番号〕(引数)

番号はUSR関数を区別するもので0から9までの数字です。省略した場合は0とみなします。この数字とDEF USR文で指定した実行開始アドレスが対応します。

引数は機械語プログラムに渡されるデータで、機械語プログラムが引数を必要としない場合でも、ダミーの引数を指定しなければいけません。

USR関数を実行するとAL、BX, DXレジスタに引数に関係するデータをセットし機械語プログラムに引き渡します。USR関数はCALL文と異なり複数のデータを引き渡すことはできません。

ALレジスタ	引数の型
BXレジスタ	FAC(フローティングアキュームレータ)中のデータを格納しているアドレス
DXレジスタ	シンボルテーブルセグメントのセグメントアドレス

それぞれのデータの形式は「第11章　データの内部構造」を参照してください。

USR関数からBASICに値を返すにはFACに引数の型と同じ型のデータを設定します。

プログラム例

USR関数の例として整数の二乗を計算する関数を示します。この例では機械語プログラムをDATA文のデータとして用意し，POKE文で書き込んでいます。

機械語プログラムではALレジスタの値を調べ整数であることを確認します。さらに与えた整数が－181から181の範囲であることを確認しています。もしこの範囲外の整数を与えた場合は－1を返します。(－181から181以外の整数を二乗すると整数の範囲外になるため)

```
100 ' 整数の二乗を求める
110 '
120 CLEAR ,&H8000       ' 機械語プログラムエリアの確保
130 DEF SEG=&H8000      ' 機械語プログラムセグメント
140 '
150 FOR PTR=0 TO 28     ' 機械語プログラムの読み込み
160   READ CODE : POKE PTR,CODE
170 NEXT
180 '
190 DEF USR=0           ' 機械語プロフラムの開始アドレス
200 '
210 INPUT A%
220   B%=USR(A%)        ' 機械語プログラムの実行
230   PRINT A%,B%
240 GOTO 210
250 '                     機械語プログラム
260 DATA &H50            :' PUSH    AX
270 DATA &H3C,&H02       :' CMP     AL,02
280 DATA &H75,&h16       :' JNE     001B
290 DATA &H8B,&h07       :' MOV     AX,WORD[BX]
300 DATA &H3D,&HB5,&H00 :' CMP     AX,00B5
310 DATA &H7F,&H0B       :' JNLE    0017
320 DATA &H3D,&H4B,&HFF :' CMP     AX,FF4B
330 DATA &H7C,&H06       :' JL      0017
340 DATA &HF7,&HE8       :' IMUL    AX
350 DATA &H89,&H07       :' MOV     WORD[BX],AX
360 DATA &HEB,&H04       :' JMPS    001B
370 DATA &HC7,&H07,&HFF,&HFF :' MOV   WORD[BX],FFFF
380 DATA &H58            :' POP     AX
390 DATA &HCF            :' IRET
400 '

RUN
? 56
 56           3136
? -78
-78           6084
? 100
 100          10000
?
```

第11章

データの内部構造

BASICは整数型、単精度実数型、倍精度実数型、文字型のデータを扱います。本章ではこれらのデータの内部形式を解読します。

メモリ中に格納されているデータをVARPTR関数などで調べることができます。

VARPTR(変数名，0)　データを格納しているメモリのオフセットアドレスを返します。

VARPTR(変数名，1)　データを格納しているメモリのセグメントアドレスを返します。

VARPTR関数の返すアドレスはそれぞれ次の場所を示します。

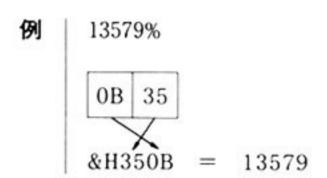
	下位アドレス← →上位アドレス
整数型	(2バイト)
単精度型	(4バイト)
倍精度型	(8バイト)
文字型	(4バイト：ストリングディスクリプタ)

□ VARPTR関数の返すアドレス

11.1 整数

整数型の数値は内部では2バイトで格納されています。整数型で表せる数値の範囲は－32768から32767までの範囲の整数です。負の数は2の補数で表します。

VARPTR関数で得たアドレスから格納されているデータは次のようにして実際の数値に変換することができます。

例

11.2 単精度実数

単精度実数は内部では4バイトで格納されています。4バイトのうち3バイトが仮数部と符号、1バイトが指数部になります。仮数部の上位バイトのビット7は符号ビットになっており、正なら0、負なら1になります。

指数部は &H80 をベースとして、−127から＋127の値が入ります。0は特別で単精度実数が0であることを示します。

実際のメモリ中には下位アドレスから仮数部の下位バイト、中位バイト、上位バイト、指数部の順に格納します。

VARPTR 関数で得たアドレスから格納されているデータは次のようにして実際の数値に変換することができます。

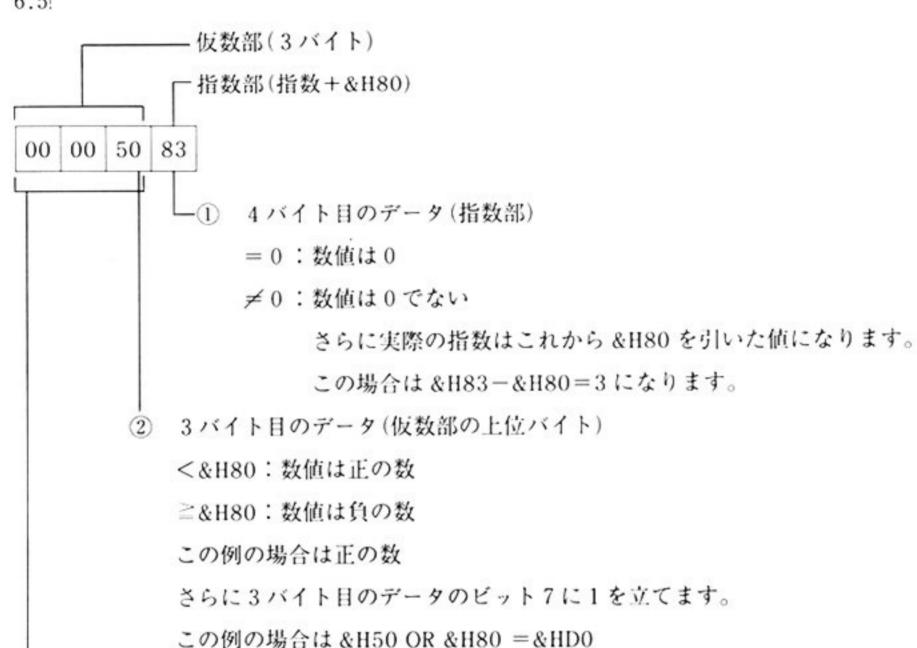
例　6.5!

③ 仮数部を並べ換え指数部と組合せます。

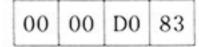
この場合は&HD0, &H00, &H00となり2進数では

0.110100000000000000000000

これに指数部を組合わせて

$0.110100000000000000000000 \times 2^{3}$

＝110.1となり10進数で6.5になります。

④ この結果に②の符号をつけたものが実際の数値になります。

ちなみに−6.5の内部表現は次のようになります。

11.3　倍精度実数

倍精度実数は内部では８バイトで格納されています。単精度実数と同じように指数形式で表します。仮数部を７バイトで表現し、単精度実数よりも多くの有効桁を表すことができます。仮数部の上位バイトのビット7は符号ビットになっており、正なら０、負なら１になります。

指数部は &H80 をベースとして、－127から＋127の値が入ります。０は特別で倍精度実数が０であることを示します。

実際のメモリ中には下位アドレスから仮数部の下位バイト、・・・・・・、上位バイト、指数部の順に格納します。

例　6.5#

仮数部（７バイト）
指数部（指数＋&H80）

00	00	00	00	00	00	50	83

（８バイト）

①　８バイト目のデータ（指数部）
　＝０：数値は０
　≠０：数値は０でない
　　さらに実際の指数はこれから &H80 を引いた値になります。
　　この場合は &H83－&H80＝3 になります。

②　７バイト目のデータ（仮数部の上位バイト）
　＜&H80：数値は正の数
　≧&H80：数値は負の数
　この例の場合は正の数
　さらに７バイト目のデータのビット７に１を立てます。
　この例の場合は &H50 OR &H80 ＝&HD0

③　仮数部を並べ換え指数部と組合せます。
　この場合は &HD0, &H00, &H00, &H00, &H00, &H00 となり２進数では
　0.110100000000000000000000000000000000000000000000
　これに指数部を組合わせて
　0.110100000000000000000000000000000000000000000000×２^{3}
　＝110.1で10進数で6.5になります。

④　この結果に②の符号をつけたものが実際の数値になります。

実数の誤差

小数部を含む実数は、値によっては正確な２進数で表すことができないために誤差を含むことがあります。

例

10進数0.1を２進数で表すと次のような循環小数になります。

0.1＝(0.11001100・・・・・・・・・・$\times 2^{-3}$)

したがって0.1！(単精度表現)は

CD	CC	4C	7D

のように２進数で、仮数部の小数点以下25桁目を０捨１入した値になります。

また0.1＃(倍精度表現)は

CD	CC	CC	CC	CC	CC	4C	7D

のように２進数で、仮数部の小数点以下57桁目を０捨１入した値になります。

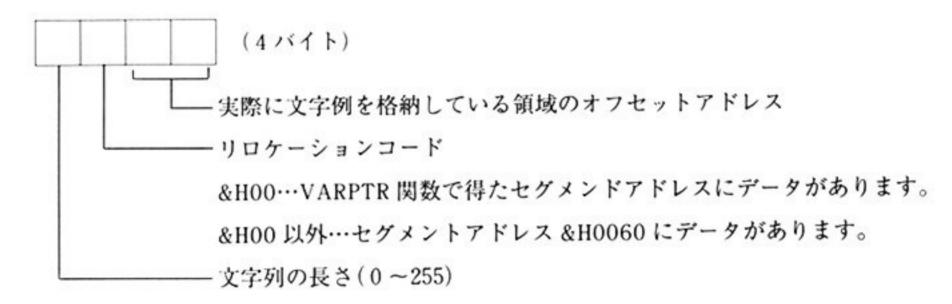
また単精度実数を CDBL 関数などで倍精度実数に型変換する場合は仮数部の下位４バイトに０を補うだけです。このため、数値の精度は単精度実数と同じであることに注意してください。１！＝１＃は真ですが0.1！＝0.1＃は偽です。

CDBL(0.1！)

00	00	00	00	CD	CC	4C	7D

このように実数には誤差を含みますので、数値の比較を行う場合には注意が必要です。倍精度が必要なときは定数も倍精度実数型定数を使う必要があります。

11.4 文字列

ストリングディスクリプタと呼ばれる４バイトの領域から実際の文字列のあるアドレスを参照します。

日本語文字列の構造

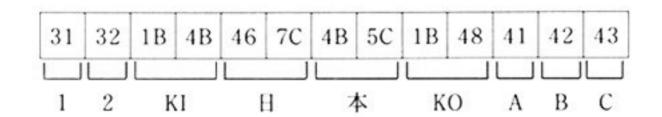
漢字などの２バイト文字は英数字(１バイト文字)２文字分の領域を必要とします。

また２バイト文字の先頭には1バイト文字と区別するための KI コードを挿入し、２バイト文字から１バイト文字に戻るために KO コードを挿入します。

　　ＫＩコード：&H1B、&H4B の２バイトのデータ

　　ＫＯコード：&H1B、&H48 の２バイトのデータ

２バイト文字を含んだ文字列を比較したり、文字数を求めたりする場合は、この KI コード、KO コードも考慮されることに注意します。

例　A$="12日本ABC"

11.5 サンプルプログラム

データの内部表現を調べるためのプログラム例を次に示します。サブルーチンになっていますので使用する場合は、それぞれのデータをX%, X!, X#またはX$にセットしてGOSUB文を実行してください。

整数の内部表現	GOSUB	*DISP. INT
単精度実数の内部表現	GOSUB	*DISP. SNG
倍精度実数の内部表現	GOSUB	*DISP. DBL
文字列の内部表現	GOSUB	*DISP. STR

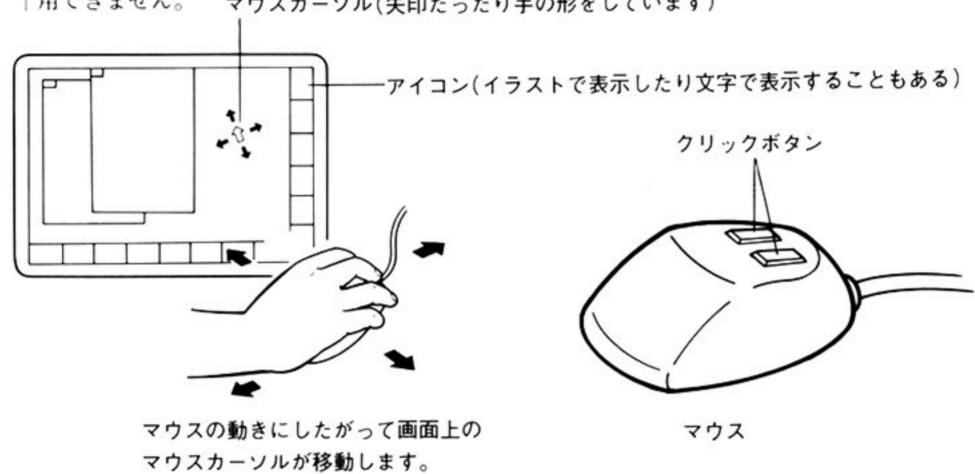
```
50000 *DISP.INT
50010   DEF SEG=VARPTR(X%,1)
50020   ADDRESS=VARPTR(X%,0)
50030   FOR OFFSET=0 TO 2-1
50040     PRINT RIGHT$("0"+HEX$(PEEK(ADDRESS+OFFSET)),2);"  ";
50050   NEXT OFFSET
50060   PRINT ":  ";X%
50070   RETURN
50080 '
50090 *DISP.SNG
50100   DEF SEG=VARPTR(X!,1)
50110   ADDRESS=VARPTR(X!,0)
50120   FOR OFFSET=0 TO 4-1
50130     PRINT RIGHT$("0"+HEX$(PEEK(ADDRESS+OFFSET)),2);"  ";
50140   NEXT OFFSET
50150   PRINT ":  ";X!
50160   RETURN
50170 '
50180 *DISP.DBL
50190   DEF SEG=VARPTR(X#,1)
50200   ADDRESS=VARPTR(X#,0)
50210   FOR OFFSET=0 TO 8-1
50220     PRINT RIGHT$("0"+HEX$(PEEK(ADDRESS+OFFSET)),2);"  ";
50230   NEXT OFFSET
50240   PRINT ":  ";X#
50250   RETURN
50260 '
50270 *DISP.STR
50280   DEF SEG=VARPTR(X$,1)
50290   ADDRESS=VARPTR(X$,0)
50300   LENGTH=PEEK(ADDRESS)
50310   IF PEEK(ADDRESS+1)<>0 THEN DEF SEG=&H60
50320   ADDRESS=PEEK(ADDRESS+2)+PEEK(ADDRESS+3)*&H100
50330   FOR OFFSET=0 TO LENGTH-1
50340     PRINT RIGHT$("0"+HEX$(PEEK(ADDRESS+OFFSET)),2);"  ";
50350   NEXT OFFSET
50360   PRINT ":  ";X$
50370   RETURN
```

第12章

マウス

マウスは、入力装置の一つで、その特徴ある形と動きからマウス(ねずみのこと)と呼ばれています。実際に使うときには、マウスを手で軽く握ってテーブルの上を滑らせ、必要に応じてマウスのボタンを押したり離したりします。

BASICでは直接マウスの制御を行うことはできませんが、機械語サブルーチンのユーティリティとしてマウスを制御するためのプログラムを用意しています。したがって、機械語サブルーチンを呼び出す要領で簡単に使用することができます。

注意 ここで使用可能なマウスは本体のマウスコネクタに接続するマウスです。一般にシリアルマウスとよばれるRS-232Cインターフェイスに接続するマウスはこのユーティリティでは使用できません。

12.1 マウスドライバ

マウスを制御するための機械語プログラムは日本語Disk BASICユーティリティディスクの中に

mouse. cod

というファイル名で登録してあります。

注意 FILESコマンドでファイルを調べるとmouse*codと表示されるものです。mouseとcodの間の*は機械語プログラムを表す識別の印です。

これがマウスを制御するための機械語プログラム(マウスドライバ)です。このマウスドライバをメモリに読み込むことによって、マウスが使用可能になります。

マウスドライバはマウスを制御するための16個の機能を持っています。BASICプログラムから、この機能をコールしてマウスを自由自在に使いこなすことができます。

12.2 画面モード

マウスを使用する場合はグラフィック画面のモードを次のいずれかに設定しなければなりません。

高解像カラーモード（横640ドット×縦400ドット）

標準カラーモード（横640×縦200ドット）

白黒モードでは使用できませんので注意してください。高解像カラーモードと標準カラーモードでは画面上での座標の取り方、マウスカーソルの形状が異なります。

注意 「マウスカーソル」とは、カーソル基準点の位置を画面に表示するための図形で、一般的には指先（手）、矢印、十字（クロス）などの図形が使われます。
本章では、「マウスカーソル」のことを「カーソル」と記述しています。

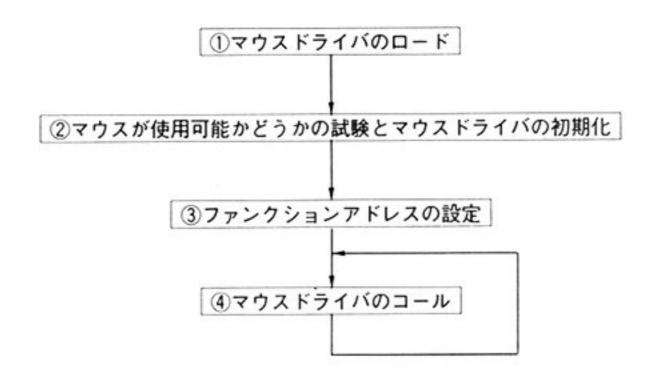
マウスの位置はディスプレイ座標の座標値で示します。

	座標のとり方	カーソル形状
高解像カラーモード	(0, 0)　(639, 0) (0, 399)　(639, 399)	16ドット × 32ドット カーソル基準点(1, 1)
標準カラーモード	(0, 0)　(639, 0) (0, 199)　(639, 199)	16ドット × 16ドット カーソル基準点(1, 1)

注意 座標値はディスプレイ座標系で表し(Dx, Dy)で示します。
カーソルの位置はカーソル基準点の座標値を返します。

12.3 マウスドライバの使用方法

マウスドライバ内の16個の機能をBASICから使用する手順を説明します。マウスドライバの持つ16個の機能については「12.4　マウスドライバの機能」で説明します。

マウスドライバを使用するときは、次の手順に従います。

①、②、③は、実際にマウスドライバをコールする前の準備としてプログラムの最初に一回だけ実行します。

12.3.1 マウスドライバのロード

マウスドライバはmouse.codというファイル名でユーティリティディスクに収められています。これをメモリの機械語領域にロードします。マウスドライバは約4.3KBの容量がありますので、これに見合う十分な機械語領域を確保してください。

マウスドライバをロードするためのサンプルプログラムは次のようになります。

```
100 CLEAR，&H8000
110 DEF SEG=&H8000
120 BLOAD "mouse．cod"
```

12.3.2 マウスドライバの初期化

マウスドライバをロードし、マウスが使用可能になるようにマウスドライバを初期化します。この処理はマウスドライバの100番地をコールして行います。

マウスドライバ初期化ルーチン

パラメータ	0	：標準カラーモード
	3	：高解像カラーモード
リターン	－1	：初期化正常終了
	0	：エラー

マウスハードウェアに問題がなければ、このルーチンは、マウスドライバの内部のファンクションエントリアドレスを割り込みベクタテーブルの&H33番目に用意します。(INT　33H)

マウスドライバを初期化するためのサンプルプログラムは次のようになります。

```
130  SETUP=&H100
140  MODE%=3                    高解像カラーモード(標準カラーモードの場合は0)
150  CALL SETUP(MODE%)
160  IF MODE%=0 THEN PRINT "マウス使用不可"  :END
```

12.3.3 ファンクションアドレスの設定

割り込みベクタテーブルの&H33番目のエントリアドレスからファンクションアドレスを読み出して実際のマウスドライバのアドレスを設定します。

マウスドライバはこのアドレス+3番地をコールします。

マウスドライバのアドレスを設定するためのサンプルプログラムは次のようになります。

```
170  DEF SEG=0
180  ADDR=PEEK(&H33*4)+PEEK(&H33*4+1)*&H100
190  MOUSE=ADDR+3
200  DEF SEG=&H8000
```

12.3.4 マウスドライバのコール

マウスドライバは、次の形式でコールします。

```
300  CALL MOUSE(AX%, BX%, CX%, DX%, ES%)
```

ここでMOUSEは190行でマウスドライバのアドレスを設定した変数です。したがってアドレスが設定されていれば変数名は何でも構いません。

AX%, BX%, CX%, DX%, ES%はマウスドライバへ引き渡す引数(パラメータ)の並びです。変数名は何でも構いませんが、必ず整数型変数を5つ並べます。これらのパラメータはマウスドライバのレジスタAX, BX, CX, DX, ESにそれぞれ対応します。

コールする際のパラメータについては、次のことに注意してください。

- マウスドライバのファンクションによっては、一部のパラメータしか使用していませんが、マウスドライバをコールする際は、必ず5つのパラメータの変数名を指定してください。
- パラメータは必ず整数型変数です。
- 結果を返す機能をコールしたとき、結果は対応する位置の引数(整数型変数)にセットされます。

12.4 マウスドライバの機能

マウスドライバはCALL文の第1パラメータ(前ページの例ではAX%)の値により、次の16種類の機能を実行します。

以下、これらの機能の内容を説明しますが、パラメータはすべてレジスタ(AX, BX, CX, DX, ES)で表現していますので、使用する際にはCALL文に引き渡すパラメータに順に対応させてください。(ここではAX%, BX%, CX%, DX%, ES%)

ファンクション一覧表

機能(10進数)	説　　明
0	初期設定
1	カーソル表示
2	カーソル消去
3	カーソル位置とボタン状態の読み取り
4	カーソル位置の設定
5	左ボタンを押した回数の読み取り
6	左ボタンを離した回数の読み取り
7	右ボタンを押した回数の読み取り
8	右ボタンを離した回数の読み取り
9	マウスカーソルの形状の設定
10	未定義(使用不可)
11	マウス移動距離の読み取り
12	ユーザー定義サブルーチンの設定
13	未定義(使用不可)
14	未定義(使用不可)
15	ミッキー／ドット比の設定
16	水平方向のカーソル移動範囲の設定
17	垂直方向のカーソル移動範囲の設定
18	カーソル表示画面の設定

初期設定(機能０)

〔引数〕 AX＝0

〔結果〕 AX＝ステータス

－1：正常終了

0：エラー

〔説明〕 マウスドライバの初期化を行い、次のような設定を行います。

マウスカーソルの表示：表示しない

カーソル位置　　　：ディスプレイの中心の位置

高解像カラーモード　：(319，199)

標準カラーモード　　：(319，99)

カーソルの形状　　：システム設定値(左上向きの手形)

カーソルの表示画面　：プレーン２

カーソルの基準点　：(1，1)

カーソル移動範囲　：ディスプレイ全体

水平方向　　　　　：0～639

垂直方向　　　　　：高解像カラーモード：0～399

標準カラーモード　：0～199

ミッキー／ドット比　：水平方向、垂直方向ともに８

マウスの割り込み周期：8 msec

注意　カーソル位置、カーソルの表示画面、カーソル移動範囲の設定は、この機能を最初に実行したときの初期値です。それ以外の場合は、現在の状態が保存されます。
ミッキーはマウスの移動距離の単位。1ミッキー＝約1/100インチ

マウスカーソルの表示(機能１)

〔引数〕 AX＝1

〔結果〕 なし

〔説明〕 カーソルを画面に表示します。マウスを動かすと、カーソルもそれに従って画面上を移動します。機能２のカーソル・オフを実行するまで表示します。

マウスカーソルの消去(機能２)

〔引数〕 AX＝2

〔結果〕 なし

〔説明〕 カーソルを画面に表示しません。ただし、マウスを動かしている間は、内部的にカーソル位置も移動しています。

カーソル位置とボタン状態の読み取り(機能3)

[引数] AX＝3

[結果] AX＝左ボタンの状態

0：押していない

－1：押している

BX＝右ボタンの状態

0：押していない

－1：押している

CX＝カーソルの水平位置

0～639

DX＝カーソルの垂直位置

高解像カラーモード：0～399

標準カラーモード　：0～199

[説明] 現在のカーソル位置と､そのときの左右のボタンの状態を読み取ります。カーソル位置はディスプレイ座標の水平座標、垂直座標を返します。ボタンの状態は押しているのか、押していないかの情報を返します。

マウスカーソルの位置指定(機能4)

[引数] AX＝4

CX＝カーソルの水平位置(0～639)

DX＝カーソルの垂直位置

(高解像カラーモード：0～399)

(標準カラーモード：0～199)

[結果] なし

[説明] 指定した位置にカーソルを移動します。カーソルの移動範囲を越えて指定した場合は、最大位置または最小位置に設定します。

左ボタンを押した回数の読み取り(機能5)

[引数] AX＝5

[結果] AX＝現在の左ボタンの状態

0：押していない

－1：押している

BX＝左ボタンを押した回数

CX＝左ボタンを最後に押したときのカーソルの水平位置

DX＝左ボタンを最後に押したときのカーソルの垂直位置

説明 左ボタンを押した状況を返します。

(1) 左ボタンを押しているのかいないのかの情報

(2) 前回、この機能を実行してから、今回実行するまでの間に左ボタンを押した回数。本機能を実行した後、押した回数は0にリセットします。

(3) 最後に左ボタンを押したときのカーソル位置

左ボタンを離した回数の読み取り（機能6）

引数 AX＝6

結果 AX＝現在の左ボタンの状態

0：押していない

－1：押している

BX＝左ボタンを離した回数

CX＝左ボタンを最後に離したときのカーソルの水平位置

DX＝左ボタンを最後に離したときのカーソルの垂直位置

説明 左ボタンを離した状況を返します。

(1) 左ボタンを押しているのかいないかの情報

(2) 前回、この機能を実行してから、今回実行するまでの間に左ボタンを離した回数。本機能を実行した後、離した回数は0にリセットします。

(3) 最後に左ボタンを離したときのカーソル位置

右ボタンを押した回数の読み取り（機能7）

引数 AX＝7

結果 AX＝現在の右ボタンの状態

0：押していない

－1：押している

BX＝右ボタンを押した回数

CX＝右ボタンを最後に押したときのカーソルの水平位置

DX＝右ボタンを最後に押したときのカーソルの垂直位置

説明 右ボタンを押した状況を返します。

(1) 右ボタンをが押しているのかいないかの情報

(2) 前回、この機能を実行してから、今回実行するまでの間に右ボタンを押した回数。本機能を実行した後、押した回数は0にリセットします。

(3) 最後に右ボタンを押したときときのカーソル位置

右ボタンを離した回数の読み取り(機能８)

〔引数〕 AX＝８

〔結果〕 AX＝現在の右ボタンの状態

　０：押していない

－１：押している

BX＝右ボタンを離した回数

CX＝右ボタンを最後に離したときのカーソルの水平位置

DX＝右ボタンを最後に離したときのカーソルの垂直位置

〔説明〕 右ボタンを離した状況を返します。

(1) 右ボタンを押しているのかいないかの情報

(2) 前回、この機能を実行してから、今回実行するまでの間に右ボタンを離した回数。本機能を実行した後、離した回数は０にリセットします。

(3) 最後に右ボタンを離したときのカーソル位置

マウスカーソルの形状の設定(機能９)

〔引数〕 AX＝９

BX＝カーソル基準点の水平位置(０～15)

CX＝カーソル基準点の垂直位置

(高解像カラーモード：０～31)

(標準カラーモード：０～15)

ES：DX＝カーソルの形を定義するデータブロックの先頭アドレス

〔結果〕 なし

〔説明〕 カーソルの基準点および形状を設定します。

カーソル基準点はカーソルの位置を示す座標で、カーソルの位置はカーソル基準点の位置を返します。指定できる範囲は次のとおりです。

水平位置　　：０～15

垂直位置　　：高解像カラーモード：０～31

　　　　　　　標準カラーモード：０～15

カーソルの形状は、あらかじめ形状を示すデータを次の順番で定義しておき、そのデータブロックの先頭アドレスを指定します。データブロックの大きさは高解像カラーモードで64バイト、標準カラーモードで32バイトのデータが必要です。

データブロックの定義

高解像カラーモード

下の図は高解像カラーモードの場合のカーソルの形状のシステム設定値です。各ビットの"1"がドットの表示を、"0"が消去を意味します。したがって、初期設定時にはこの形(左上向きの手形)のカーソルを表示します。カーソル基準点は(1, 1)です。

```
   1バイト目        2バイト目
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0
0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0
0 0 1 1 0 1 1 0 0 0 0 0 0 0 0 0
0 0 0 1 1 0 1 1 0 0 0 0 0 0 0 0
0 0 0 1 1 0 1 1 0 1 0 0 0 0 0 0
0 0 1 0 1 1 0 1 1 0 1 0 0 0 0 0
0 0 1 0 1 1 1 1 1 1 0 1 0 0 0 0
0 0 1 1 0 1 1 1 1 1 1 1 1 0 0 0
0 0 1 1 0 1 1 1 1 1 1 1 1 0 0 0
0 0 1 1 0 1 1 1 1 1 1 1 1 0 0 0
0 0 1 1 1 1 1 1 1 1 1 1 1 0 0 0
0 0 0 1 1 1 1 1 1 1 1 1 1 0 0 0
0 0 0 1 1 1 1 1 1 1 1 1 1 0 0 0
0 0 0 0 1 1 1 1 1 1 1 1 1 0 0 0
0 0 0 0 0 1 1 1 1 1 1 1 1 0 0 0
0 0 0 0 0 0 1 1 1 1 1 1 1 0 0 0
0 0 0 0 0 0 0 1 1 1 1 1 1 1 0 0
0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 0
0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 0
0 0 0 0 0 0 0 0 0 1 1 1 1 1 0 0
0 0 0 0 0 0 0 0 0 1 1 1 1 0 0 0
0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
                   最後のバイト
```

このカーソル形状の場合のデータブロックは次のようになります。

00, 00, 60, 00, 60, ・・・・・00

標準カラーモード

下の図は標準カラーモードのカーソルの形状のシステム設定値です。カーソル基準点は(1, 1)です。

```
   1バイト目        2バイト目
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 1 1 0 1 1 0 0 0 0 0 0 0 0
0 0 0 0 1 1 0 1 1 0 1 0 0 0 0 0
0 1 1 0 1 1 1 0 1 1 0 1 0 0 0 0
0 1 1 0 1 1 1 1 1 1 1 1 1 0 0 0
0 0 1 1 1 1 1 1 1 1 1 1 1 0 0 0
0 0 1 1 1 1 1 1 1 1 1 1 1 0 0 0
0 0 0 0 1 1 1 1 1 1 1 1 1 0 0 0
0 0 0 0 0 0 1 1 1 1 1 1 1 1 0 0
0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
                   最後のバイト
```

このカーソル形状の場合のデータブロックは次のようになります。

00, 00, 60, 00, 30, ・・・・・00

マウスの移動距離の読み取り(機能11)

〔引数〕 AX＝11

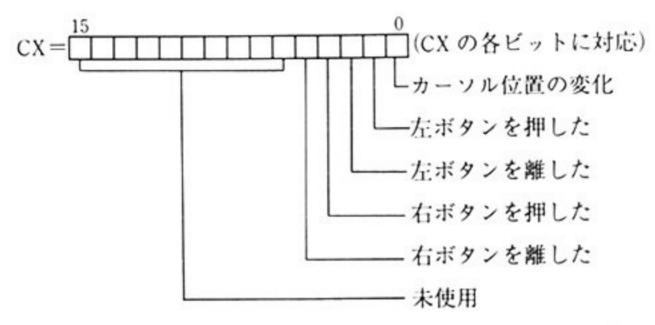
〔結果〕 CX＝水平方向の移動距離(−32768〜32767)

DX＝垂直方向の移動距離(−32768〜32767)

〔説明〕 この機能を前回実行したときからのマウスの移動距離を返します。この機能を実行後、移動距離(CX, DX)は0になります。

移動距離は水平方向では右方向に正の値が、垂直方向では下方向に正の値を加えます。

ユーザー定義サブルーチンの設定(機能12)

〔引数〕 AX＝12

(各ビットが1のときON, 0のときオフになります。)

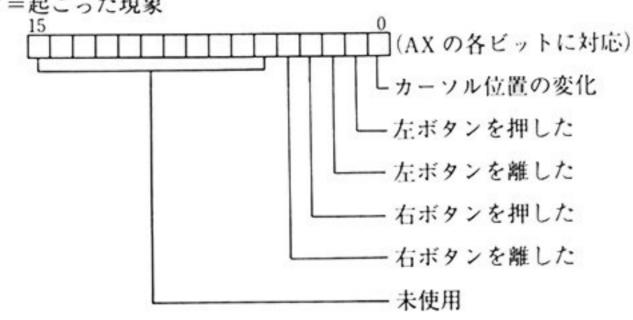
ES：DX＝ユーザー定義サブルーチンのアドレス

〔結果〕 なし

〔説明〕 CXレジスタで指定した現象が起こったときに、あらかじめ定義しておいたユーザー定義サブルーチンを実行します。CXレジスタに複数のビットをオンにした場合は、そのうち1つ以上の現象が起こったときに実行します。

定義したサブルーチンはfarコールで呼ばれるため、必ずfarリターンしてください。

またユーザー定義サブルーチンを実行する前に、各レジスタを次のように設定します。

AXレジスタ＝起こった現象

BLレジスタ＝左ボタンの状態

0：押していない

−1：押している

BH レジスタ＝右ボタンの状態

0 ：押していない

－1 ：押している

CX レジスタ＝カーソル位置の水平座標

DX レジスタ＝カーソル位置の垂直座標

ミッキー/ドット比の設定（機能15）

引数 AX＝15

CX＝水平方向のミッキー/ドット比

DX＝垂直方向のミッキー/ドット比

結果 なし

説明 カーソルを8ドット動かすのに必要なマウスの移動距離をミッキーの単位で設定します。100ミッキーは約1インチ(2.54cm)です。これによりマウスの感度を変えることができます。

水平方向のカーソル移動範囲の設定（機能16）

引数 AX＝16

CX＝水平方向のカーソル移動範囲の最小値

DX＝水平方向のカーソル移動範囲の最大値

結果 なし

説明 カーソルの水平方向の移動範囲を設定します。移動範囲は、その最小値と最大値を設定することにより決まります。カーソル基準点はこの範囲内を移動します。CX パラメータの値がDX パラメータの値より大きい場合は、DX パラメータを最小値、CX パラメータを最大値とします。

CX、DX パラメータの値の範囲は0から639までです。

垂直方向のカーソル移動範囲の設定（機能17）

引数 AX＝17

CX＝垂直方向のカーソル移動範囲の最小値

DX＝垂直方向のカーソル移動範囲の最大値

結果 なし

説明 カーソルの垂直方向の移動範囲を設定します。移動範囲は、その最小値と最大値を設定することにより決まります。カーソル基準点はこの範囲内を移動します。CX パラメータの値がDX パラメータの値より大きい場合は、DX パラメータが最小値、CX パラメータが最大値になります。

ファンクション16，17によってカーソルの移動範囲を変更して、カーソルの位置が移動範囲外になった場合は、マウスドライバはマウスカーソルを移動範囲内の端に移動します。

CX、DX パラメータの値の範囲は次のとおりです。

高解像カラーモード：0～399

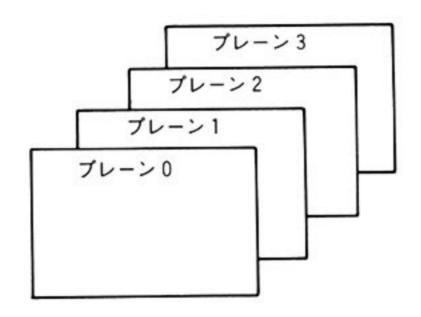
標準カラーモード：0～199

カーソル表示画面の設定(機能18)

引数 AX＝18

BX＝カーソルの表示画面

結果 なし

説明 カーソルの表示画面を設定します。カーソルの色は表示画面のパレットで設定された色に従います。

グラフィック用VRAMは内部的には4つに分割して使用しておりディスプレイ画面にはこの4つの画面を合成したものを表示しています。

分割した画面はプレーン0，プレーン1，プレーン2，プレーン3という名称で区別します。

プレーン3については4096色中16色モードで使用します。

マウスカーソルはこのプレーンの1枚を選択して表示しています。

マウスカーソルをどのプレーンに表示するのかを指定するのがこのファンクションです。

各画面の初期値は

プレーン0：青(パレット1)

プレーン1：赤(パレット2)

プレーン2：緑(パレット4)

プレーン3：灰色(パレット8)

12.5 サンプルプログラム

このプログラムはマウスドライバの機能のうち

- 初期設定
- マウスカーソルの表示
- カーソル位置とボタン状態の読み取り
- 水平方向のカーソル移動範囲の設定
- 垂直方向のカーソル移動範囲の設定
- カーソル表示画面の設定

の5つの機能を使用しています。

注意 このプログラムは、4096色表示が可能なアナログRGBディスプレイを使用しないと正しく動作しません。

プログラムを実行すると、画面にR(赤)、G(緑)、B(青)のパレットを表示し、それぞれの色の組み合せによって、どのような色が表示されるのかが一目でわかります。

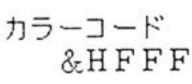
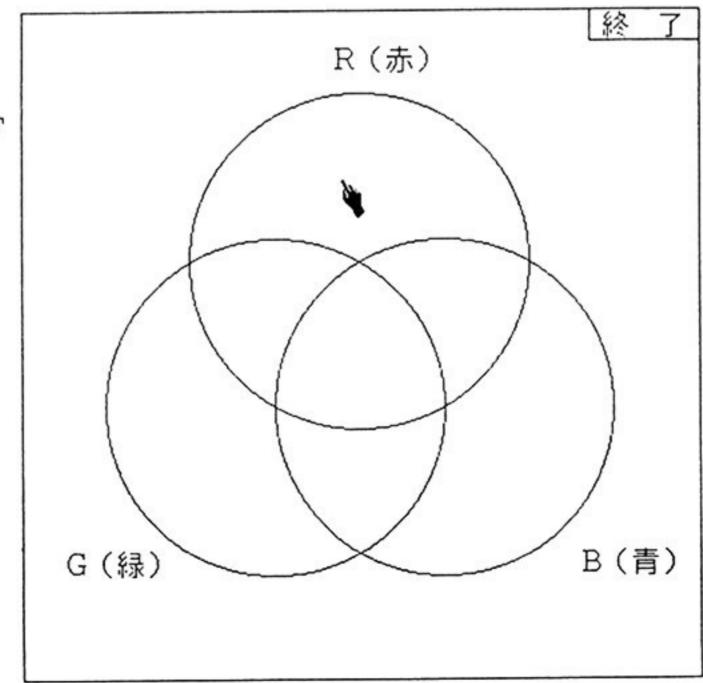
マウスカーソルをR，G，Bいづれかのパレットに移動してクリックボタンを押します。左ボタンを押すと輝度が上がり、右ボタンを押すと輝度が下がります。R，G，Bすべての色を組み合わせたときのカラーコードを画面の左に表示します。

終了する場合は、画面上の終了と書かれた領域にカーソルを移動し、クリックボタンを押します。

注意　このプログラムは、4096色表示が可能なアナログRGBディスプレイを使用しないと正しく動作しません。また、ディップSW1-8をONにしてBASICを起動してください。

```
100 ' マウスサンプルプログラム
110 '
120 ' 光の３原色（ＲＧＢ）の輝度を変化させ４０９６色を表示
130 '
140 SCREEN 3,0 : COLOR ,,,,2     ' 高解像カラーモード、４０９６色
150 WIDTH 80,20 : CONSOLE ,,0,1 ' ファンクションキー消去
160 CLS  3
170 CLEAR ,&H8000  : DEF SEG=&H8000 ' 機械語プログラム領域の確保
180 BLOAD "mouse.cod"             ' マウスドライバの読み込み
190 '
200 SETUP=&H100                   ' マウスドライバ初期化
210 SCRNP%=3  : CALL SETUP(SCRNP%)
220 IF SCRNP%=0 THEN PRINT "マウス使用不可能" : END
230 '
240 DEF SEG=0                     ' 実行開始アドレスの設定
250 ADR=PEEK(&H33*4)+PEEK(&H33*4+1)*256
260 MOUSE=ADR+3
270 DEF SEG=&H8000
280 '
290 COLOR                         ' 画面作成
300 LINE(120,0)-(520,399),15,BF,0
310 X=320 : Y=150 : R=100
320 LOCATE 38,1  : COLOR 2 : PRINT "R（赤）"
330 LOCATE 18,16 : COLOR 4 : PRINT "G（緑）"
340 LOCATE 56,16 : COLOR 1 : PRINT "B（青）" : COLOR 7
350 LOCATE 58,0 : PRINT "終  了"
360 LINE(456,0)-(520,19),15,B
370 CIRCLE(X,Y),R,15
380 CIRCLE STEP(-R/2,R*SQR(3)/2),R,15
390 CIRCLE STEP(R,0),R,15
400 '
410 D=R/2
420 PAINT(X,Y-D),2,15                '赤(R)
430 PAINT(X-D,Y),6,15                '黄色(R+G)
440 PAINT(X,Y+D),7,15                '白(R+G+B)
450 PAINT(X+D,Y),3,15                '紫(R+B)
460 PAINT(X-R/2-D,Y+R*SQR(3)/2),4,15 '緑(G)
470 PAINT(X+R/2+D,Y+R*SQR(3)/2),1,15 '青(B)
480 PAINT(X,Y+R+D),5,15              '水色(G+B)
490 '
500 AX%=0 : BX%=0 : CX%=0 : DX%=0 : ES%=0
510 CALL MOUSE(AX%,BX%,CX%,DX%,ES%)  '初期設定
520 AX%=18 : BX%=3 : CALL MOUSE(AX%,BX%,CX%,DX%,ES%)
530                                  'カーソルの移動範囲設定
540 AX%=16 : CX%=120 : DX%=520 : CALL MOUSE(AX%,BX%,CX%,DX%,ES%)
550 AX%=17 : CX%=0   : DX%=399 : CALL MOUSE(AX%,BX%,CX%,DX%,ES%)
560 AX%=1  : CALL MOUSE(AX%,BX%,CX%,DX%,ES%) ' カーソル表示
570 R=15 : G=15 : B=15 : QUIT=(1=0)
580 '
590 *MAINLOOP                        '色を１６段階に変更
```

```
600   LOCATE 0,2 : PRINT "カラーコード "
610   LOCATE 4,3 : COLOR 7 : PRINT "&H";
620   COLOR 4 : PRINT AKCNV$(HEX$(G));
630   COLOR 2 : PRINT AKCNV$(HEX$(R));
640   COLOR 1 : PRINT AKCNV$(HEX$(B))  : COLOR 7
650   AX%=3 : CALL MOUSE(AX%,BX%,CX%,DX%,ES%)
660   IF AX% THEN UD=1  : GOSUB *MOUSEON : GOTO *CHECK
670   IF BX% THEN UD=-1 : GOSUB *MOUSEON
680 *CHECK
690   IF NOT QUIT THEN *MAINLOOP
700   AX%=2 : CALL MOUSE(AX%,BX%,CX%,DX%,ES%)
710   CLS 3
720 END
730 '
740 *MOUSEON
750   IF CX%<456 OR CX%>520 THEN 770
760   IF DX%>19 THEN 770 ELSE QUIT=(1=1) : RETURN  '終了
770   C=POINT(CX%-1,DX%-1) 'カーソル位置の色を調べる
780   IF C=1 THEN B=B+UD : GOTO *LIMIT
790   IF C=2 THEN R=R+UD : GOTO *LIMIT
800   IF C=4 THEN G=G+UD : GOTO *LIMIT
810   RETURN
820 *LIMIT                        '色は0からFまで
830   IF R<0  THEN R=0
840   IF R>15 THEN R=15
850   IF G<0  THEN G=0
860   IF G>15 THEN G=15
870   IF B<0  THEN B=0
880   IF B>15 THEN B=15
890 '                              カラーコード作成
900   C=B                : COLOR=(1,C)
910   C=R*16             : COLOR=(2,C)
920   C=R*16+B           : COLOR=(3,C)
930   C=G*16*16          : COLOR=(4,C)
940   C=G*16*16+B        : COLOR=(5,C)
950   C=(G*16+R)*16      : COLOR=(6,C)
960   C=(G*16+R)*16+B    : COLOR=(7,C)
970   FOR I=1 TO 2000 : NEXT : RETURN  'リターン
980 '
```

第13章

ディスクのファイル管理

本章では簡単なディスクの構造やBASICのファイル管理の方法などについて説明します。

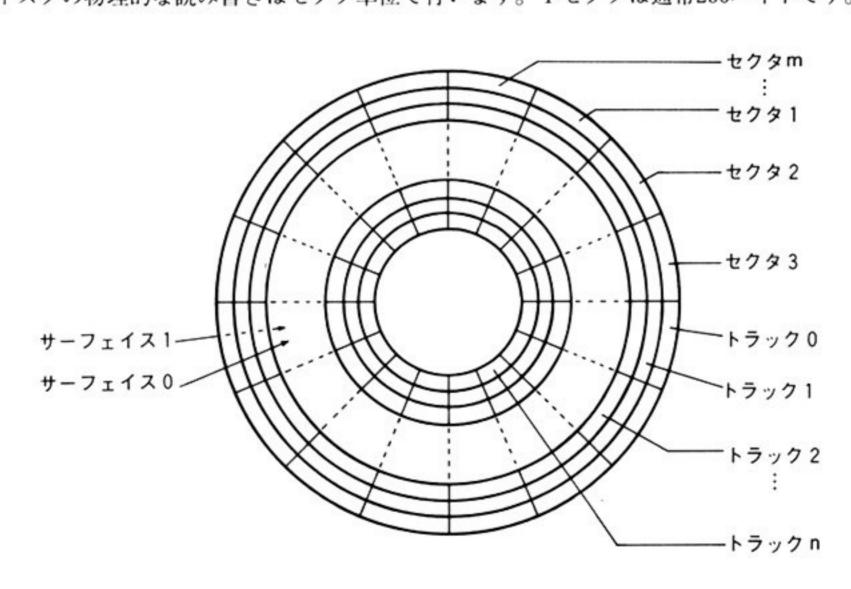
13.1　ディスクのフォーマット

新しいフロッピーディスクをBASICで使用するためには、フォーマットと呼ばれる初期化を行う必要があります。このフォーマットは使用するシステムにより異なるため、他のシステム(MS-DOS)などで使用していたフロッピーディスクをBASICで使用する場合もフォーマットを行ってください。フォーマットはディスク上にデータの読み書きを行うための領域の区分けを行います。そして区分けした領域を単位としてデータを書き込んだり、読み出したりします。

ディスクのフォーマットはユーティリティソフトBMENUの「フォーマット」を使用します。詳細については「第2部　BASICユーティリティ」を参照してください。

ハードディスクを使用する場合も同様にフォーマットを行います。ただしハードディスクのフォーマットは装置全体を初期化する物理フォーマットとハードディスク上のBASIC領域内を初期化する論理フォーマットに分けられます。物理フォーマットはBMENUの「ハードディスク・メンテナンス」で、論理フォーマットは「フォーマット」で行います。

ディスクはサーフェイスと呼ばれるディスクの面と、それぞれの面に同心円上に書き込まれたトラック、そのトラックをいくつかに分割したセクタから構成されます。

ディスクの物理的な読み書きはセクタ単位で行います。1セクタは通常256バイトです。

フロッピーディスクの例

BASICで使用可能なディスクの仕様は次の通りです。

ディスクの種類	サーフェイス番号	トラック番号	セクタ番号
両面高密度ディスク(2HD) (5.25インチ、3.5インチ) 8インチ倍密度ディスク	0～1	0～76	1～26
両面倍密度倍トラックディスク(2DD) (5.25インチ、3.5インチ)	0～1	0～79	1～16
20MBハードディスク	0～7	0～307	1～33
40MBハードディスク	0～7	0～613	1～33

注意　両面高密度ディスクと8インチ倍密度ディスクのサーフェイス0トラック0は1セクタ128バイトでフォーマットしています。
ハードディスクの値は論理的なもので物理的な仕様と異なります。

13.2 ディスクのファイル管理

BASICで扱うファイルは、ファイル名によってBASICが管理しているため、直接ユーザーがこの内容を知る必要はありません。ただしDSKI$, DSKO$文などで直接ディスクにアクセスする場合などには必要な情報です。

BASICが実際にディスク上のファイルをアクセスするには、データを記録している場所の物理アドレス(サーフェイス番号、トラック番号、セクタ番号)をドライブに知らせる必要があります。これらのファイルの位置を管理するのにBASICはディレクトリとFATという2つの管理テーブルを使っています。ただし、実際のファイルの管理はセクタより大きいクラスタと呼ばれる単位で行っています。クラスタの大きさはディスクの種類によって異なります。

ディスクの読み書きはセクタ単位で行いますが、使用領域の確保、開放はクラスタ単位でしか行えません。すなわち、あるクラスタのセクタを一つでも使用すれば、そのクラスタのほかのセクタを使用していなくても、最初に使用したファイル以外のファイルでは使用できないことになります。

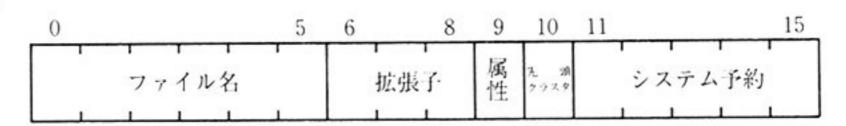
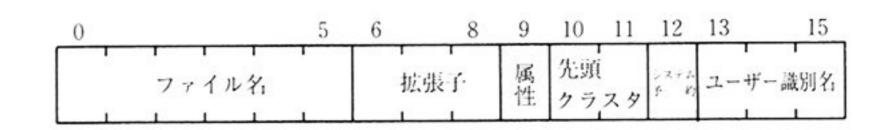
13.2.1 ディレクトリ

ディレクトリの中には、そのディスクにセーブされているすべてのファイルのファイル名やファイルの属性などの情報が入っています。

一つのファイルに対するディレクトリの情報をディレクトリエントリと呼び、16バイトで構成されています。BASICは、ディレクトリと後述のFATによってファイルの物理的な格納場所の対応付けを行います。

ディレクトリで管理できるファイルの数は、ディスクの種類によって異なります。

ディレクトリは1つのファイルに対して16バイトの情報を持っています。16バイトの内容は次のとおりです。

フロッピーディスク

ハードディスク

ファイル名

ファイル名の欄にはSAVE文やOPEN文などで指定したファイルの名前が左詰めで入ります。余った欄は空白(&H20)で埋められます。同様に拡張子の欄には指定したファイル名拡張子の名前が入ります。拡張子を指定しなかった場合はすべて空白でディレクトリが空いていることを示します。またファイル名の先頭が&hFFであれば、このディレクトリエントリより後ろのディレクトリエントリはまだ使用されていないことを意味します。

ファイルの属性

属性はファイルの仕様を表すもので、各ビットごとに次の意味を持ちます。

7　　　　0

ファイルの種類	0	SAVE文やOPEN文で作成した一般のファイル
	1	BSAVE文で作成したファイル
未使用	システム予約	
ファイルの書き込み禁止	0	書き込み可能
	1	書き込み禁止。SET文でP指定するとこのビットが1になります
プロテクトセーブ	0	修正可能なプログラムファイルです。通常は0になります。
	1	プロテクトセーブしたプログラムファイルです。このプログラムに対して修正を行おうとしてもエラーになります。SAVE文でP指定をするとこのビットが1になります。
書き込みチェック（リード・アフタ・ライト）	0	書き込み時のチェック行わない
	1	書き込みチェックを行う。SET文でR指定をするとこのビットが1になります。
プログラムファイル	0	アスキー形式のプログラムファイル。SAVE文でA指定をするとこのビットが0になります。
	1	バイナリ形式のプログラムファイル

先頭クラスタ番号

ファイルの先頭クラスラの番号が入ります。クラスタ番号の大きさはフロッピーディスクでは1バイト、ハードディスクでは2バイトになります。FATを参照してください。

システム予約

&HFFで埋められています。

ユーザー識別名

ハードディスクを接続していると、BASICを起動するときに「User identifier」と尋ねてきます。これに対して答えた3文字のデータがこの欄に書き込まれます。BASICはユーザー識別名の一致したファイルだけを処理の対象にします。

例　起動時にユーザー識別名として「ABC」を入力すると、ハードディスク上に作成するすべてのファイルにユーザー識別名として「ABC」を書き込みます。これらのファイルは起動時にユーザー識別名として別の3文字を入力したときには使用できないファイルになります。例えばFILESコマンドを実行しても画面に表示されません。

ディレクトリ領域の確認(2HDフロッピーディスクの例)

```
h]^D1,0,23,1…機械語モニタの CTRL + D コマンドでディレクトリをダンプします。
Drive=01 Surface=00 Track=0023 Sector=01

0000 65 67 64 69 63 6D 64 69 63 00 47 FF FF FF FF FF        egdicmdic G
0010 65 67 64 69 63 75 64 69 63 00 45 FF FF FF FF FF        egdicudic E
0020 65 67 7A 69 70 6D 64 69 63 00 41 FF FF FF FF FF        egzipmdic A
0030 65 67 7A 69 70 75 64 69 63 00 38 FF FF FF FF FF        egzipudic 8
0040 42 4D 45 4E 55 20 20 20 20 80 37 FF FF FF FF FF        BMENU     _7
0050 62 6D 65 6E 75 20 20 20 20 80 33 FF FF FF FF FF        bmenu     _3
0060 42 4D 45 4E 55 20 41 55 54 80 32 FF FF FF FF FF        BMENU AUT_2
0070 42 4D 45 4E 55 20 42 4B 50 80 30 FF FF FF FF FF        BMENU BKP_0
0080 62 6D 65 6E 75 20 62 6B 70 01 2F FF FF FF FF FF        bmenu bkp /
0090 42 4D 45 4E 55 20 43 50 59 80 2E FF FF FF FF FF        BMENU CPY_.
00A0 62 6D 65 6E 75 20 63 70 79 01 2C FF FF FF FF FF        bmenu cpy ,
00B0 42 4D 45 4E 55 20 44 43 4D 80 2B FF FF FF FF FF        BMENU DCM_+
00C0 42 4D 45 4E 55 20 44 43 31 80 29 FF FF FF FF FF        BMENU DC1_)
00D0 42 4D 45 4E 55 20 44 43 32 80 28 FF FF FF FF FF        BMENU DC2_(
00E0 62 6D 65 6E 75 20 64 63 64 01 27 FF FF FF FF FF        bmenu dcd '
00F0 62 6D 65 6E 75 20 64 63 6A 01 1F FF FF FF FF FF        bmenu dcj
h]
```


13.2.2　FAT

ファイルとクラスタ

ディスクの物理的な読み書きはセクタ単位で行われますが、実際のファイルへの割り当てはクラスタと呼ばれる単位で行います。クラスタの大きさはディスクの種類によって異なります。

ディスクの種類	クラスタサイズ
5インチ(3.5インチ)2HD	26セクタ(1トラック/1クラスタ)
5インチ(3.5インチ)2DD	16セクタ(1トラック/1クラスタ)
20MBハードディスク	33セクタ(1トラック/1クラスタ)
40MBハードディスク	66セクタ(2トラック/1クラスタ)

ファイルへの割り当てをクラスタ単位で行うため、あるクラスタの先頭のセクタを使用すれば、そのクラスタの残りのセクタを使用していなくても、ほかのファイルでそのセクタを使用することはできないことになります。

FATとクラスタ

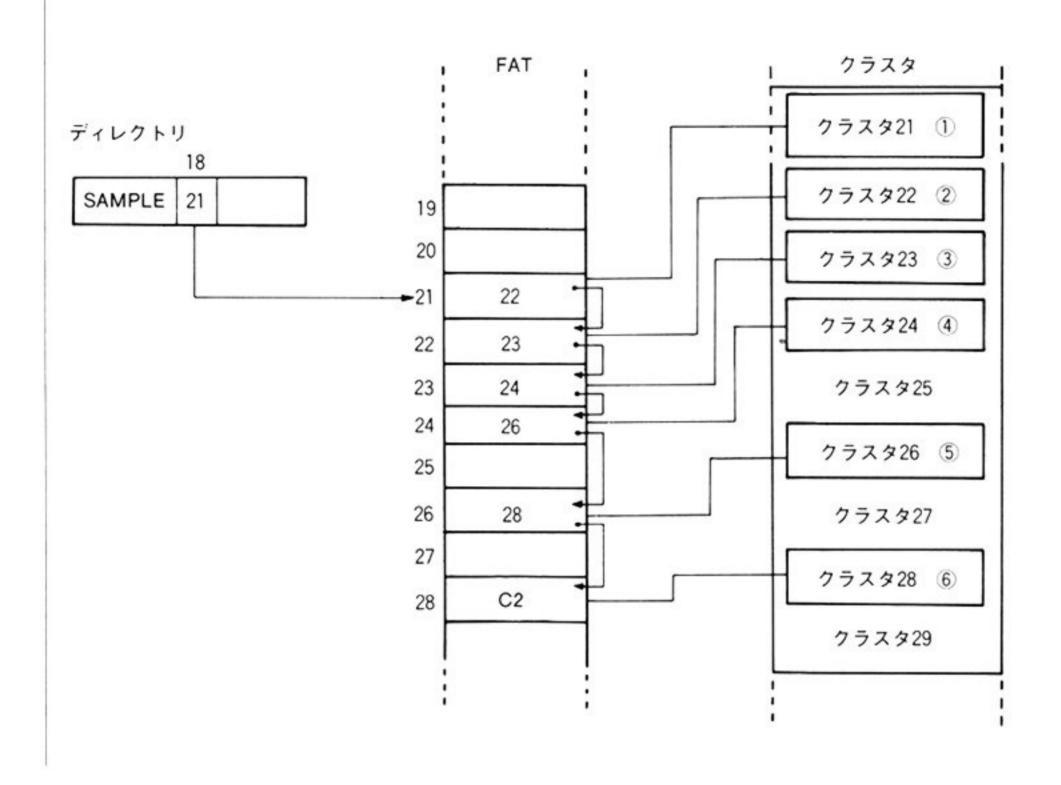
FATはディスク上のすべてのクラスタの使用状態(そのクラスタを使用しているか未使用か)を管理し、それぞれのファイルがどのクラスタをどの順序で使っているかを管理します。FATはディスク上のすべてのクラスタごとに、フロッピーディスクは1バイト、ハードディスクは2バイトのデータを持っています。これらの1バイトあるいは2バイトのデータをクラスタのエントリといいます。このデータの値&h00～&hFF(ハードディスクの場合は&h0000～&hFFFF)によってクラスタの使用状態が表され、これらの意味は次のとおりです。

BASICのシステムで予約されているクラスタに対応するエントリには&HFE(ハードディスクは&hFFFE)が、使用されていないクラスタに対応するエントリには&hFF(ハードディスクは&hFFFF)が書き込まれています。

クラスタの種類 ＼ ディスクの種類	FATの値				
	ファイルの途中(クラスタが続く)	ファイルの終了(クラスタの最後)	未使用クラスタ	不良クラスタ	システム予約クラスタ
両面高密度倍トラックディスク(2HD)	00～&H99	&HC0～&HDAのいずれか	&HFF	—	&HFE
両面倍密度倍トラックディスク(2DD)	00～&H9F	&HC0～&HD0のいずれか	&HFF	—	&HFE
20MB/40MBハードディスク	000～&H99B	&H4000～&H4021*のいずれか	&HFFFF	&HFFFD	&HFFFE

*40MBハードディスクの場合は&H4042

ファイルの先頭クラスタはディレクトリエントリの「先頭クラスタ」に書き込まれています。このクラスタに対応するクラスタエントリには2番目のクラスタ番号が格納されています。さらに2番目のクラスタ番号に対応するクラスタエントリには3番目のクラスタ番号、3番目のクラスタ番号に対応するクラスタエントリには4番目のクラスタ番号というようにつながっていき、最後のクラスタ番号に対応するエントリには、そのクラスタ内で実際に使用しているセクタ数が書き込まれます。

例 "SAMPLE"というファイルが、最初にクラスタ21を使用し、続いて22、23、24、26、最後に28を使っています。クラスタ28ではセクタを２つ使用しています。

FAT は、そのドライブを最初にアクセスしたときに読み込まれます。その後、ファイルの内容に変化があると FAT を書き直します。

フロッピーディスクには、同じ内容の FAT を３箇所に書き込んでいます。これは FAT が重要なデータであるため、もし破壊してしまったときに復旧できるように用意されています。

13.2.3 ハードディスクのフォーマット

標準フォーマットと拡張フォーマット

ハードディスクのフォーマットには 20MB までのハードディスクをフォーマットする標準フォーマットと、40MB のハードディスクをフォーマットする拡張フォーマットがあります。

参考 機種によっては 20MB のものでも拡張フォーマットできるものもあります。この場合でも 20MB ハードディスクは標準フォーマットするようにしてください。ハードディスクを BASIC だけで使用する場合は問題ありませんが、ほかのアプリケーションと分割して使用する場合には問題が起きます。

拡張フォーマットをサポートしているシステムは拡張フォーマットしたハードディスクでも標準フォーマットしたハードディスクでも読み書きすることができます。しかし、標準フォーマットのみサポートしているシステムでは拡張フォーマットされたハードディスクを読み書きすることができません。

標準フォーマットのみサポートしているシステム

日本語 DiskBASIC ver1.0

日本語 MS-DOS ver.2.××

拡張フォーマットをサポートしているシステム

日本語 DiskBASIC ver2.0

日本語 DiskBASIC ver3.0(本システム)

日本語 MS-DOS ver.3.××

アプリケーションソフトには一般的に MS-DOS ver2.11 が組み込まれており、この MS-DOS は拡張フォーマットをサポートしていません。したがって、このアプリケーションソフトと BASIC を使用する場合は、標準フォーマットをするようにしてください。このとき BASIC の領域を MS-DOS よりさきに確保するようにしてください。そうしないと BASIC の領域が認識されないことがあります。

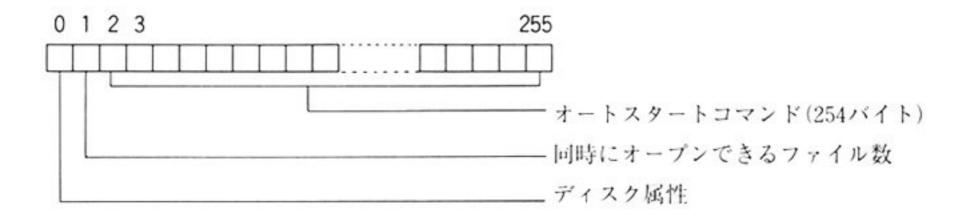
40MB のハードディスクを使用する場合は自動的に拡張フォーマットになります。

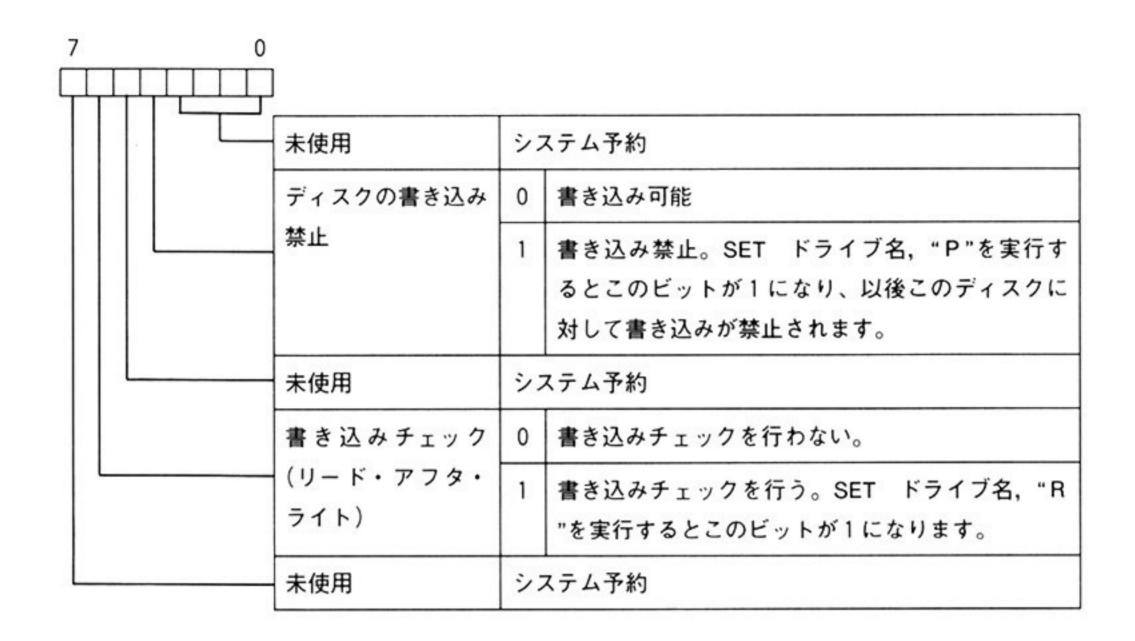
標準フォーマットのクラスタサイズは33セクタ(1トラック)ですが、拡張フォーマットでは66セクタ(2トラック)に増やされています。

標準フォーマットでは BASIC などシステムの領域は必ずひとつだけです。これに対して拡張フォーマットでは同じ BASIC の領域を複数設定できます。複数設定できますが BASIC で一度に扱えるのは1つの領域だけです。

また、拡張フォーマットではハードディスクからの起動時にスタートアッププログラムが実行され、どのシステムを起動するのかを選ぶことができます。

13.3 ID

ディスクにはディスクの内容を識別したり、オートスタートで実行されるコマンドを記録した ID と呼ばれる1セクタの領域があります。1セクタは256バイトで ID は次のような内容を持ちます。

ディスク属性

ディスク属性はそのディスクの使用環境を決定するもので各ビットごとに次の意味を持ちます。

同時にオープンできるファイル数

この欄には同時にオープンできるファイル数を設定します。この欄に255(&hFF)が書き込まれていると BASIC 起動時に次のようなメッセージを表示します(初期設定)。

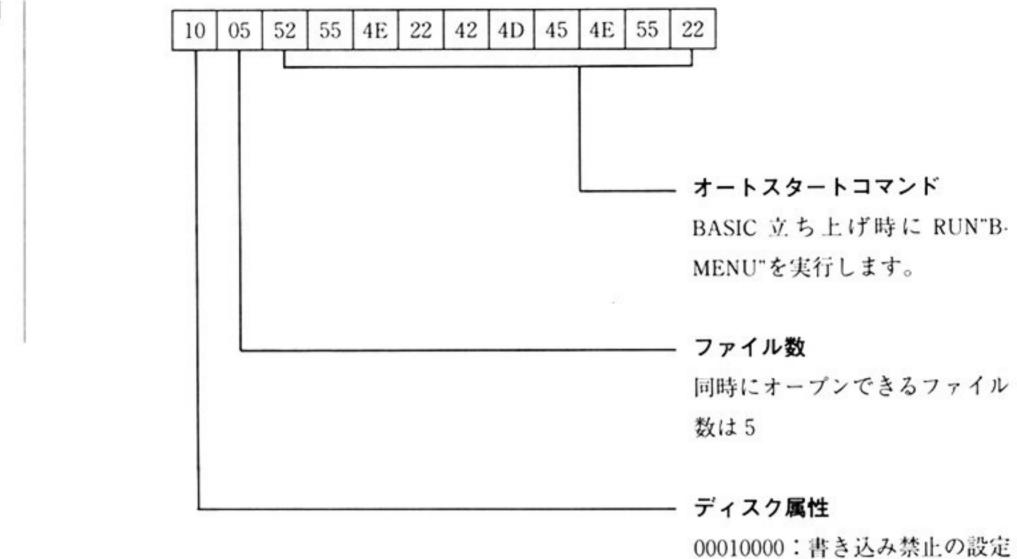
```
How many files?(0－15)
```

この欄に0から15までの数値を書き込むとこのメッセージは表示せず、ここで設定した数までのファイルを同時にオープンできることになります。

オートスタートコマンド

オートスタートコマンドはBASIC起動時に自動的に実行するBASICの命令を書き込む領域です。ここに書き込んだ命令をあたかもキーボードから入力したように実行します。

この領域の先頭が0であれば無効になります。

例

BMENU の「オートスタート設定」で入力したファイル数とオートスタートコマンドはこの ID に書き込まれます。

13.4 ディスクの仕様

DSKF 関数を使用して、ディスクの仕様を調べることができます。

関数	機能
DSKF(ドライブ名,0)	トラック番号の最大値
DSKF(ドライブ名,1)	1 トラック当りのセクタ数
DSKF(ドライブ名,2)	サーフェイス番号の最大値
DSKF(ドライブ名,3)	1 トラック当りのクラスタ数(フロッピーディスク 20MB ハードディスク) 1 クラスタ当りのトラック数(40MB ハードディスク)
DSKF(ドライブ名,4)	ディスク当りのクラスタ数
DSKF(ドライブ名,5)	ディレクトリのあるトラック番号
DSKF(ドライブ名,6)	1クラスタ当りのセクタ数
DSKF(ドライブ名,7)	FAT の開始セクタ番号
DSKF(ドライブ名,8)	FAT の終了セクタ番号
DSKF(ドライブ名,9)	FAT の数
DSKF(ドライブ名,10)	ディスク属性のセクタ番号

注意 ハードディスクの場合は、媒体のすべてをBASICが使用し、不良セクタが一つもない場合の値です。また、サーフェイス番号、トラック数も論理的な値であり、実際のハードディスクの物理的な仕様とは異なります。

仕様＼ディスク	両面高密度ディスク (2HD)		両面倍密度倍トラックディスク (2DD)	20MB ハードディスク (標準フォーマット)		40MB ハードディスク (拡張フォーマット)	
サーフェイス番号	0～1		0～1	0～7		0～7	
トラック番号	0～76		0～79	0～307		0～613	
セクタ番号	1～26		1～16	1～33		1～33	
クラスタサイズ	26セクタ		16セクタ	33セクタ		66セクタ	
クラスタ数	154		160	2480		2456	
ディレクトリ位置	サーフェイス0 トラック 35 セクタ 1～22		サーフェイス0 トラック 40 セクタ 1～12	サーフェイス0～6 トラック 155 セクタ 1～33		サーフェイス0～6 トラック 301 セクタ 1～33	
FAT位置	サーフェイス0 トラック 35 セクタ 24～26		サーフェイス0 トラック 40 セクタ 14～16	サーフェイス7 トラック 155 セクタ 2～33		サーフェイス7 トラック 301 セクタ 2～33	
ID位置	サーフェイス0 トラック 35 セクタ 23		サーフェイス0 トラック 40 セクタ 13	サーフェイス7 トラック 155 セクタ 1		サーフェイス7 トラック 301 セクタ 1	
ユーザー領域	システム	データ	データ	システム	データ	システム	データ
	サーフェイス0 トラック12～34 トラック36～76 セクタ1～26 サーフェイス1 トラック12～76 セクタ1～26	サーフェイス0 トラック1～34 トラック36～76 セクタ1～26 サーフェイス1 トラック1～76 セクタ1～26	サーフェイス0 トラック1～39 トラック41～79 セクタ1～16 サーフェイス1 トラック1～79 セクタ1～16	サーフェイス0～2 トラック3～154 トラック156～306 セクタ1～33 サーフェイス3～7 トラック2～154 トラック156～306 セクタ1～33	サーフェイス0 トラック1～154 トラック156～306 セクタ1～33 サーフェイス1～7 トラック0～154 トラック156～306 セクタ1～33	サーフェイス0～5 トラック4～300 トラック302～601 セクタ1～33 サーフェイス6～7 トラック3～300 トラック302～613 セクタ1～33	サーフェイス0～1 トラック2～300 トラック302～601 セクタ1～33 サーフェイス2～7 トラック1～300 トラック302～601 セクタ1～33

BASIC ユーティリティ

BASICユーティリティ

本章は、日本語Disk BASICユーティリティディスクに含まれているBASIC用のユーティリティソフト「BMENU」と「BTERM」「SYSSET」の使用方法について説明します。

ユーティリティソフトBMENUで行うことのできる処理は次のとおりです。

	機能の名前	内容
1	ディスクバックアップ	同じ種類のフロッピーディスクの間でバックアップコピーを行います。
2	フォーマット	フロッピーディスクのフォーマットを行います。また、ハードディスクに対しては領域確保を行います。
3	ファイルコピー	ファイル単位にコピーを行います。
4	ハードディスク・メンテナンス	ハードディスクのフォーマットと、ハードディスク上のファイルのバックアップおよびリストアを行います。
5	メモリスイッチ変更	メモリスイッチの内容を変更し、さまざまな機能の変更や追加を行います。
6	日本語入力機能メンテナンス	日本語入力機能を使いやすくするために、動作環境の変更を行います。また、日本語辞書のメンテナンス、および郵便番号辞書のメンテナンスを行います。
7	ユーザーフォント・メンテナンス	フォントが未定義である文字(ユーザー定義外字)のフォントを作成し、プログラム中でそのフォントを使えるような処理を行います。
8	拡張ハードコピー・オートスタート設定	BASICのプログラムを自動的に実行するオートスタートの機能を設定します。また、ページプリンタ用に画面のハードコピーの機能拡張を行います。

ユーティリティソフト BTERM で行う処理は次のとおりです。

1　ターミナル・モード	BASIC の動作モードをターミナルモードに切り換え、データ通信を行います。 また、ファイルのアップロード、ダウンロード、通信内容の印刷、参照、通信パラメータの設定を行います。
2　エディタ・モード	アップロード用ファイルの作成、編集。ダウンロードしたファイルの編集を行います。
3　通信パラメータ設定	データ通信を行なうための各種パラメータを設定し、設定した値をパラメータファイルに書き込みます。

ユーティリティソフト SYSSET で行う処理は次のとおりです。

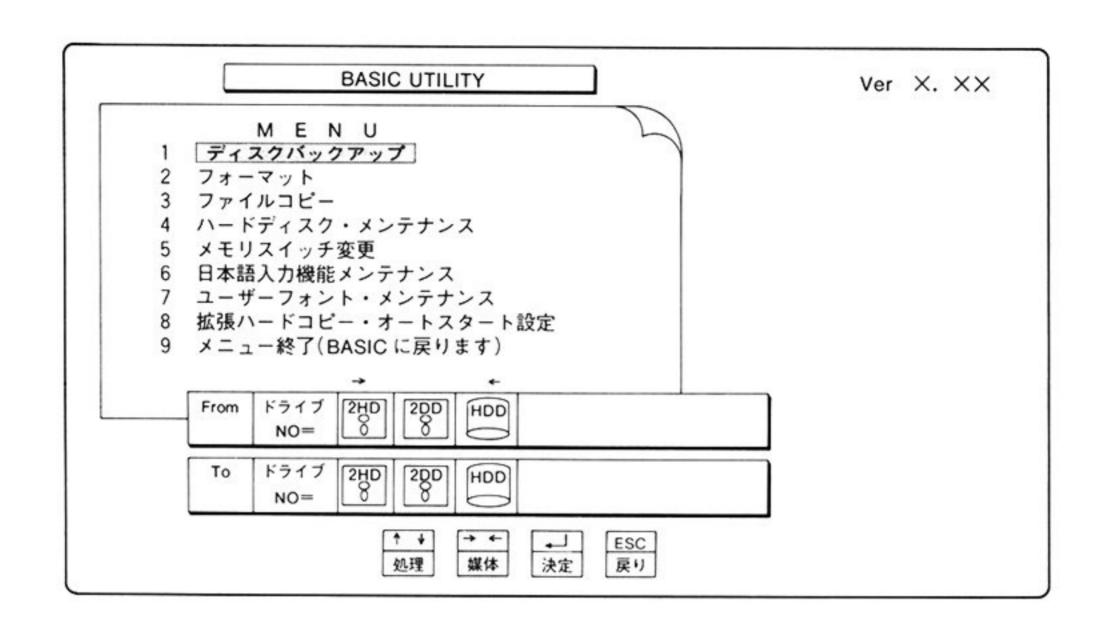
1　最小システムの作成	ディップスイッチ、メモリスイッチの設定によらず、必要最低限なシステムだけを読み込むシステムディスクを作成します。

1　BMENU の実行と終了

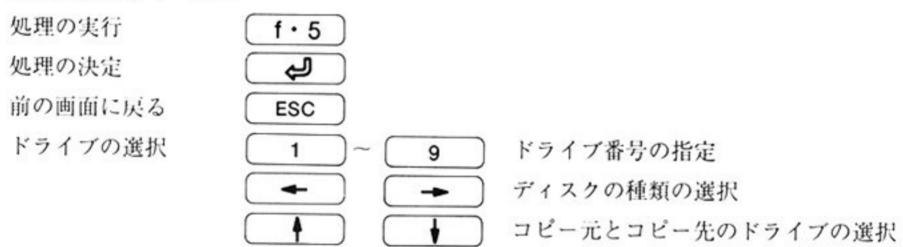
BMENU を実行するには、ディスク上に次のファイルが必要になります。特にメインプログラムは必ず必要なファイルです。

① 日本語 Disk BASIC ユーティリティディスクをドライブ1にセットし

RUN "BMENU ⏎ または RUN "bmenu ⏎

と入力します。次のようなメニューを表示します。

② 実行する処理をカーソル移動キー［↑］［↓］あるいは数字キーを使って選び、［↵］キーを押します。各機能ごとにサブメニューとキーの使用方法を表示しますので指示に従ってください。

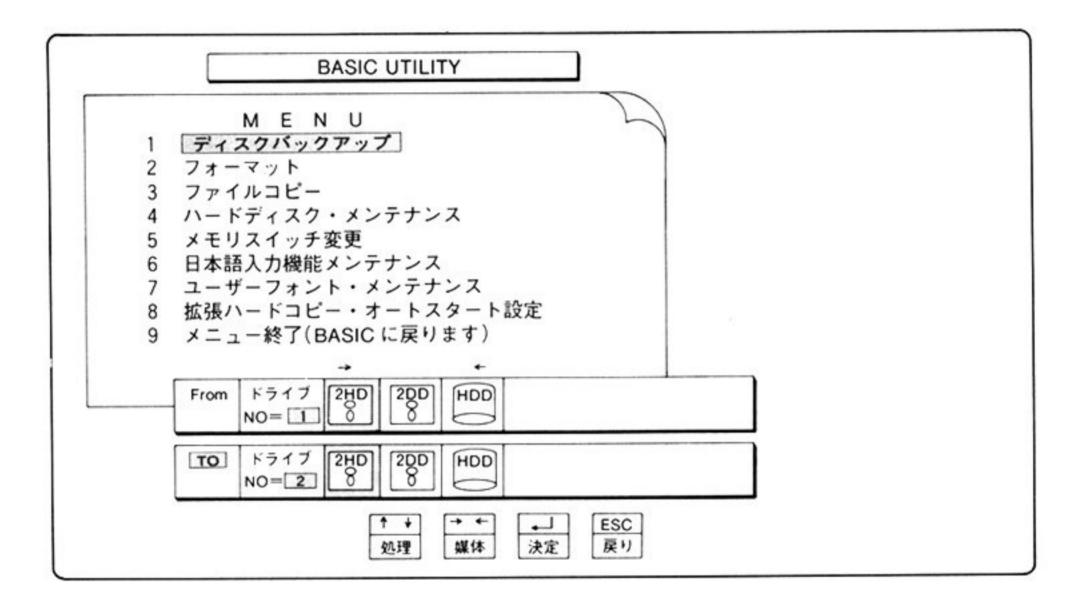
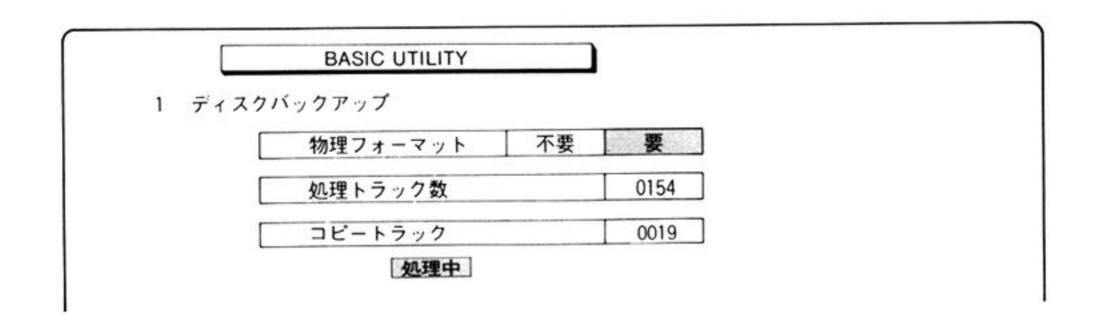
各処理共通なキー操作

処理の実行	［f・5］	
処理の決定	［↵］	
前の画面に戻る	［ESC］	
ドライブの選択	［1］～［9］	ドライブ番号の指定
	［←］［→］	ディスクの種類の選択
	［↑］［↓］	コピー元とコピー先のドライブの選択

③ 各処理から［ESC］キーを押すと一つ前の画面に戻ります。これを繰り返すと最終的にメインメニューに戻ります。メインメニューで「9　メニュー終了(BASICに戻ります)」を選び［↵］キーを押すとユーティリティの実行を終了し、コマンド入力モードになります。

注意　［ESC］キーを押して前の画面に戻る際は必ずドライブ1にユーティリティの入ったディスクをセットするようにしてください。
それぞれの機能をサブプログラムから実行しようとすると"BMENU から実行してください。"のメッセージが表示されます。必ずメインプログラム"BMENU"、"bmenu"から実行してください。

2　ディスクバックアップ

内容のまったく同じディスクを作ることをバックアップを取るといい、原本のシステムディスクのコピーを取るときなどに使用します。バックアップは同じ種類のディスクの間でしか行うことはできません。

① メインメニューから「ディスクバックアップ」にカーソル(反転表示)を合わせて ⏎ キーを押します。
② 「From」(原本のフロッピーディスクを入れるドライブ)のドライブ番号と「To」(コピーされるフロッピーディスクを入れるドライブ)のドライブ番号、ディスクの種類を選び、⏎ キーを押します。

③ コピー先のディスクが既にフォーマットしてあるものであれば「不要」を、それ以外の場合は「要」を ← → で選択し、⏎ キーを押します。
④ それぞれのドライブにフロッピーディスクをセットし、 f・5 キーを押します。処理を行わずメインメニューに戻るには ESC キーを押します。
⑤ 表示が次のように変わり、バックアップ処理を開始します。

正常に終了するとスピーカが鳴り、**処理終了**と表示します。

⑥ 処理終了後 ESC キーを押すと③の表示に戻ります。ここでフロッピーディスクを取り換えれば続けてバックアップ処理を行うことができます。③の表示で ESC キーをもう一度押すとメインメニューに戻ります。

参考 ドライブ番号に同じ番号を指定すると、原本のディスクとコピーされるディスクを交互に入れ換えて1台のフロッピーディスクドライブでバックアップを行うことができます。

```
BASIC UTILITY

1 ディスクバックアップ

| 全面コピー | システムコピー |
```

「転送元ディスク」と「転送先ディスク」の指示に従ってディスクを交換しながらディスクコピーを行います。転送先ディスクはあらかじめフォーマットしておくようにしてください。

```
| 全面コピー | システムコピー |
転送元ディスクを入れてスペースキーを押してください。
```

また、フロッピーディスクドライブを1台しか持っていない機種でシステムディスクを作成する場合は、「ディスクバックアップ」の「システムコピー」を実行します。この場合、システムをコピーするディスクはあらかじめフォーマットしておくようにしてください。

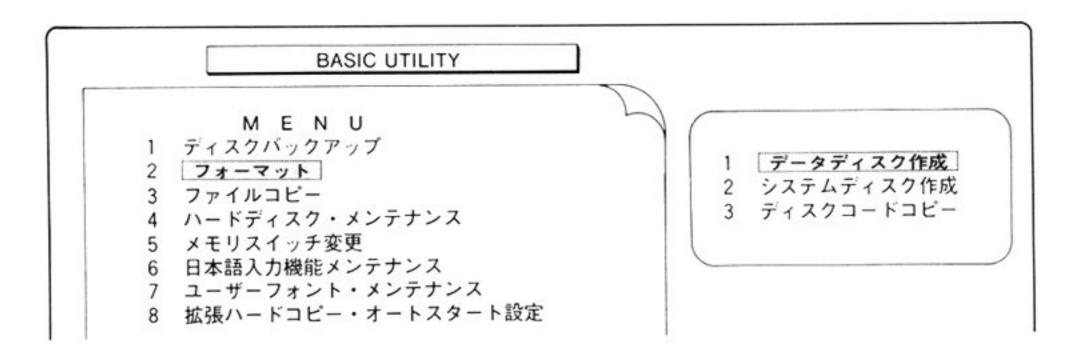
```
| 全面コピー | システムコピー |
転送元ディスクを入れてスペースキーを押してください。
```

「転送元ディスク」と「転送先ディスク」の指示に従ってディスクを交換してシステムコピーを行います。

「フォーマット」の中の「システムディスク」と「ディスクコードコピー」は2台のフロッピーディスクドライブがないと実行できません。

3　フォーマット

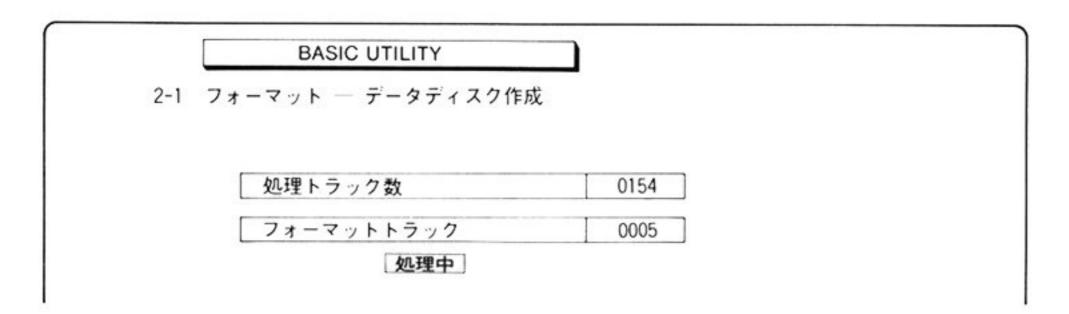
購入したばかりのディスクや、ほかのシステムで使っていたディスクを、BASICで使用できるように初期化(フォーマット)します。

① メインメニューから「フォーマット」にカーソル(反転表示)を合わせて ⏎ キーを押します。
② 表示が次のように変わります。

ディスクをデータディスクとして用いる場合は「データディスク」を、システムディスクとして使用する場合は「システムディスク」を選びます。「システムディスク」を選ぶとフォーマット終了後、システムプログラム(ディスクコード)を書き込み、BASIC起動用ディスクとして使用することができます。「ディスクコードコピー」は既にフォーマットしてあるディスクにシステムプログラムだけを書き込みます。

注意　2DD(640KB)のディスクにシステムプログラムを書き込むことはできません。
ハードディスクをフォーマット(領域確保)した後は、一度リセットをしてください。再び、BASICが起動した時点から、ハードディスクが使用できる状態になります。

3.1　フロッピーディスク

3.1.1　データディスク作成

① サブメニューで「データディスク」を選びます。
② フォーマットするディスクを入れるドライブのドライブ番号とディスクの種類を選びます。
③ ⏎ キーを押すと表示が次のように変わます。指定したドライブにディスクを入れ、f・5 キーを押すとフォーマットを開始します。

④　フォーマット処理中は

処理中

と表示し、正常に終了するとスピーカが鳴り

処理終了

と表示します。

⑤　処理終了後 [ESC] キーを押すと③の画面に戻ります。ここで、フロッピーディスクを取り換えれば、続けてフォーマットを行うことができます。③の画面で [ESC] キーをもう一度押すとメインメニューに戻ります。

3.1.2　システムディスク作成

①　サブメニューで「システムディスク作成」を選びます。

②　「From」(システムディスクを入れるドライブ)のドライブ番号とディスクの種類、「To」(フォーマットするディスクを入れるドライブ)のドライブ番号とディスクの種類を選び [↵] キーを押します。

③　それぞれのドライブにディスクをセットし、 [f・5] キーを押します。処理を行わずメインメニューに戻るには [ESC] キーを押します。

④　フォーマット処理後、システムプログラムを「From」のディスクから読み込み、「To」のディスクに書き込みます。処理中は**システム作成中**と表示します。
正常に終了するとスピーカが鳴り、**処理終了**と表示します。

⑤　処理終了後 [ESC] キーを押すと③の画面に戻ります。ここで、フロッピーディスクを取り換えれば、続けてフォーマットを行うことができます。③の画面で [ESC] キーをもう一度押すとメインメニューに戻ります。

3.1.3　ディスクコードコピー

システムディスクからシステムプログラムをディスクに書き込む処理です。フォーマットは行いません。

①　サブメニューから「ディスクコードコピー」を選び、 [↵] キーを押します。

②　「システムディスク作成」と同様に「From」のドライブと「To」のドライブを入力します。画面の表示に従ってキーを入力します。

③　「From」のドライブからシステムプログラムを読み出し「To」のドライブに書き込みます。処理中は**処理中**と表示します。
正常に終了するとスピーカが鳴り**処理終了**と表示します。

注意　「フォーマット」の中の「システムディスク」と「ディスクコードコピー」は2台のフロッピーディスクドライブがないと実行できません。1台のフロッピーディスクドライブでシステムディスクを作成する場合は、「ディスクバックアップ」の「システムコピー」を実行してください。

3.2　20MB ハードディスク(標準フォーマット)

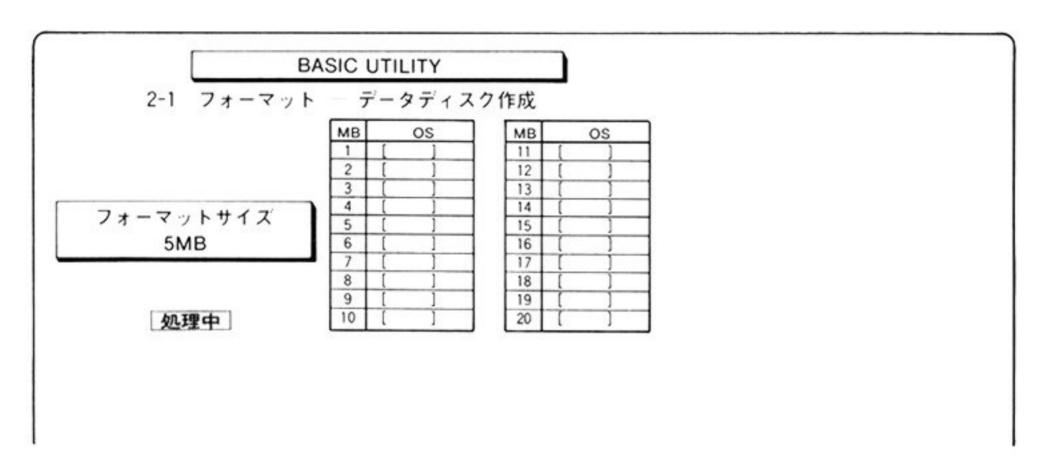
3.2.1　データディスク作成

① サブメニューで「データディスク作成」を選びます。

② ハードディスクの欄を反転表示にして、フォーマットするハードディスクのドライブ番号を入力します。

③ [↵]キーを押すと、表示が次のように変わります。

フォーマット(領域確保)する領域の大きさを入力し[↵]キーを押します。領域の大きさは、OSの欄が[　　]あるいはBASICとなっている欄の合計以下で指定します。

フォーマットを行うと、現在確保されているBASICの領域のデータは消去されてします。必要なデータがある場合は、フロッピーディスクなどに移すようにしてください。

④ [f・5]キーを押すとフォーマットを開始します。フォーマット処理中は

処理中

と表示され、正常に終了するとスピーカが鳴り、

処理終了

と表示されます。これでハードディスクのフォーマットは終了です。

⑤ [ESC]キーを押すとメインメニューに戻ります。

3.2.2　システムディスク作成

「システムディスク作成」を実行すると、ハードディスクからBASICを起動することができます。「第1章　BASICの起動と終了」の「1.10　BASICとハードディスク」を参照してください。

① 「3.1　フロッピーディスク」の「3.1.2　システムディスク作成」と同様に「From」(システムディスクをセットするドライブ)のドライブ番号と「To」(フォーマットするハードディスクドライブ)のドライブ番号を入力します。

② 「3.2.1　データディスク作成」と同じようにBASICの領域を確保しフォーマットします。

③ フォーマット処理後、システムプログラムを「From」のディスクから読み込み、ハードディスクに書き込みます。処理中は**システム作成中**と表示されます。正常に終了するとスピーカが鳴り、**処理終了**と表示されます。

④ [ESC]キーを押すと、メインメニューに戻ります。

3.2.3 ディスクコードコピー

システムディスクからシステムプログラムをハードディスクに書き込む処理です。フォーマットは行いません。

① 「3.2.2 システムディスク作成」と同様に「From」(システムディスクをセットするドライブ)のドライブ番号と「To」(システムプログラムを書き込むハードディスクドライブ)のドライブ番号を入力します。

② 「3.2.2 システムディスク作成」と同様に、システムプログラムを「From」のディスクから読み込み、ハードディスクに書き込みます。処理中は**処理中**と表示されます。正常に終了するとスピーカが鳴り、**処理終了**と表示されます。

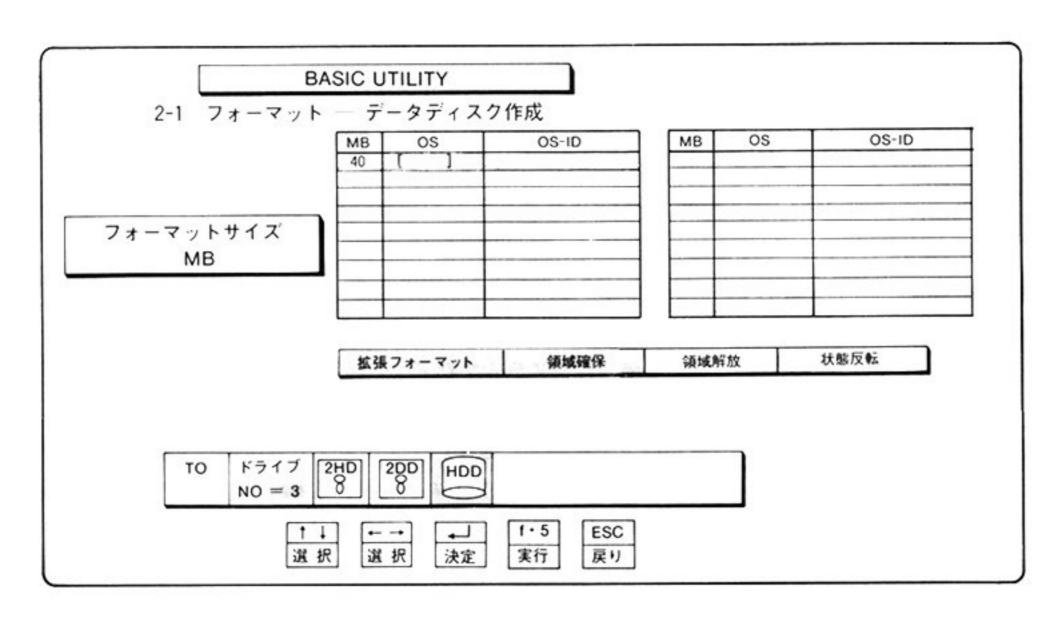
③ [ESC]キーを押すと、メインメニューに戻ります。

3.3 40MBハードディスク(拡張フォーマット)

3.3.1 データディスク作成

① サブメニューで「データディスク作成」を選びます。

② ハードディスクの欄を反転表示にして、フォーマットするハードディスクのドライブ番号を入力します。

③ [↵]キーを押すと、表示が次のように変わります。

④ カーソル移動キー[→][←]キーで「領域確保」を反転表示させます。

⑤ フォーマット(領域確保)する領域の大きさを入力し[↵]キーを押します。入力した数字が反転表示します。

領域の大きさは、OSの欄が[ ]となっている領域の数値以下を指定します。

⑥ [↵]キーを押して、実行する領域を決定します。OS−IDの欄が反転表示します。

⑦ [f・5]キーを押すとフォーマットを開始します。フォーマット処理中は

処理中

と表示され、正常に終了するとスピーカが鳴り、

処理終了

と表示されます。これでハードディスクのフォーマットは終了です。

⑧ BASICの領域を使用可能な状態にします。カーソル移動キー [→] [←] で「状態反転」を反転表示させます。カーソル移動キー [↑] [↓] で状態反転をさせたいBASIC領域を反転表示させ、[↵] キーを押します。

⑨ [f・5] キーを押すと状態反転が実行されます。OSの欄に「*」が表示されて、BASIC領域が使用可能な状態になります。OSの欄に「*」が表示されている状態で「状態変更」を実行すると「*」が消えて、使用できない状態になります。使用可能な領域には、いくつか設定したBASIC領域のうちのひとつだけしか設定できません。

⑩ [ESC] キーを押すとメインメニューに戻ります。

領域開放 必要なくなった領域をほかのシステムで使用できるように開放することができます。

① カーソル移動キー [→] [←] で「領域開放」を反転表示させます。カーソル移動キー [↑] [↓] で開放したいBASIC領域を反転表示させ、[↵] キーを押します。

② [f・5] キーを押すと領域開放が実行されます。OSの欄に [　　　] が表示されて、開放されたことが判ります。

3.3.2 システムディスク作成

「システムディスク作成」を実行すると、ハードディスクからBASICを起動することができます。「第1章　BASICの起動と終了」の「1.10　BASICとハードディスク」を参照してください。

① 「3.1　フロッピーディスク」の「3.1.2　システムディスク作成」と同様に「From」(システムディスクをセットするドライブ)のドライブ番号と「To」(フォーマットするハードディスクドライブ)のドライブ番号を入力します。

② 「3.3.1　データディスク作成」と同じようにBASICの領域の大きさを入力し [↵] キーを押します。

③ [↵] キーを押して、フォーマットする領域を決定します。OS−IDの欄が反転表示します。

④ OS−IDを入力して [↵] キーを押します。OS−IDはBASICの領域を区別するためにつける名前です。この名前のついた領域にはシステムプログラムが書き込まれています。[↵] キーだけを押したときはDisk BASIC 2.0になります。

⑤ [f・5] キーを押すとフォーマットを開始します。フォーマット処理後、システムプログラムを「From」のディスクから読み込み、ハードディスクに書き込みます。処理中は**システム作成中**と表示されます。正常に終了するとスピーカが鳴り、**処理終了**と表示されます。

⑤ 「3.2.1　データディスク作成」と同様に使用する領域は「状態変更」を実行して使用可能な状態にします。

⑥ [ESC] キーを押すと、メインメニューに戻ります。

3.3.3 ディスクコードコピー

システムディスクからシステムプログラムをハードディスクに書き込む処理です。フォーマットは行いません。

① 「3.3.2　システムディスク作成」と同様に「From」(システムディスクをセットするドライブ)のドライブ番号と「To」(システムプログラムを書き込むハードディスクドライブ)のドライブ番号を入力します。

② 「3.3.2 システムディスク作成」と同様に、システムプログラムを「From」のディスクから読み込み、ハードディスクに書き込みます。処理中は**処理中**と表示されます。正常に終了するとスピーカが鳴り、**処理終了**と表示されます。

③ 使用する領域は「状態変更」を実行して使用可能な状態にします。

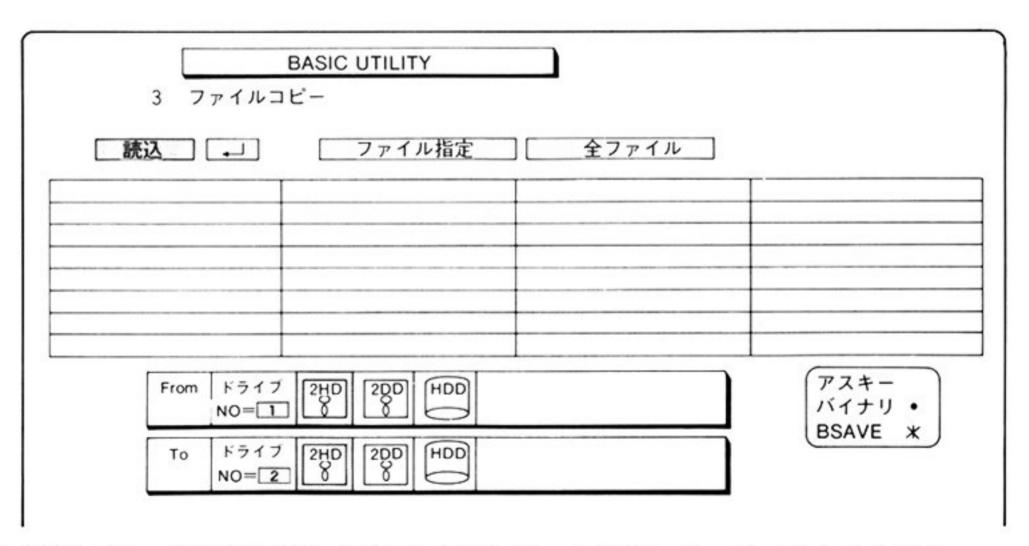
④ ［ESC］キーを押すと、メインメニューに戻ります。

4 ファイルコピー

ディスク上に記憶しているファイルをファイル単位で別のディスクにコピーします。ディスクの全ファイルをコピーすることも、指定したファイルだけをコピーすることもできます。

① メインメニューから「ファイルコピー」にカーソル(反転表示)を合わせて［↵］キーを押します。

② 「From」にコピーするファイルの入っているディスクを入れるドライブ番号とディスクの種類を、「To」にコピーを作るディスクを入れるドライブ番号とディスクの種類を入力し、［↵］キーを押します。

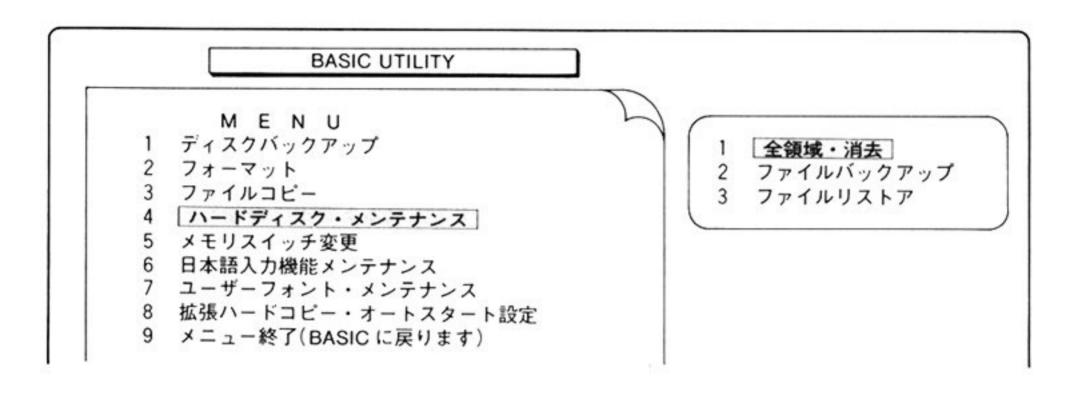
③ ファイルコピーの処理画面を表示しますので、それぞれのドライブにディスクを入れ、「読込」を反転表示して［↵］キーを押します。「From」のドライブに入っているディスクの全ファイル名を表示します。

④ 指定したファイルだけをコピーしたいときには「ファイル指定」を、ディスク上のすべてのファイルをコピーしたいときには「全ファイル」をカーソル移動キー［←］［→］で選び、［↵］キーを押します。

⑤ 「ファイル指定」の場合はコピーするファイルにカーソルを移動し［スペース］キーを押します。ファイル名が反転表示に変わり、このファイルを実行時にコピーすることを示します。もう一度［スペース］キーを押せば通常の表示になり、コピーはしません。
「全ファイル」を選択するとすべてのファイルをコピーします。

⑥ ［f・5］キーを押すと反転表示したファイルを順次コピーしていきます。コピーの終わったファイルは通常の表示に戻ります。

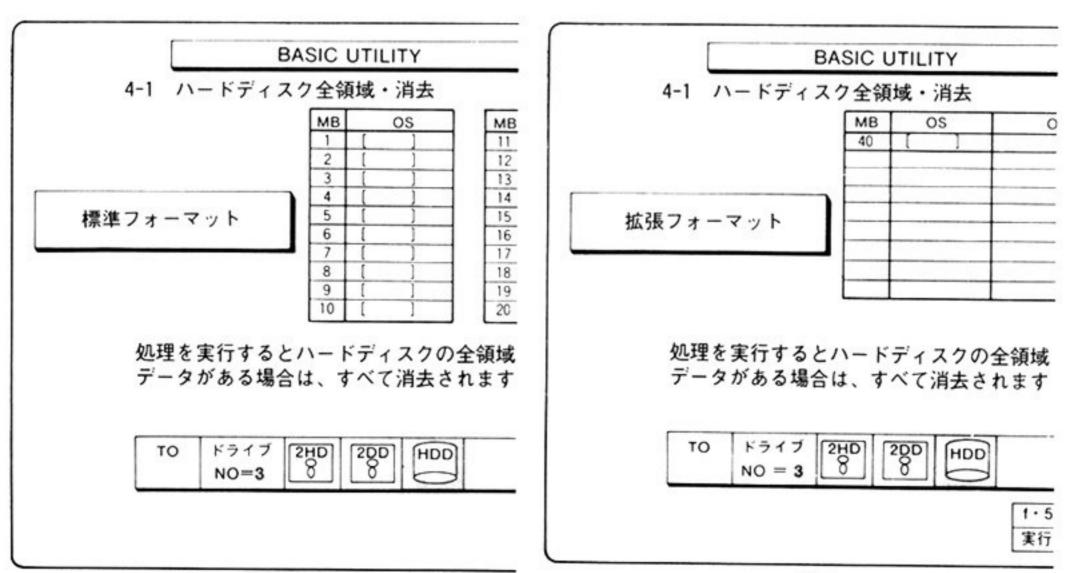
5　ハードディスクメンテナンス

ハードディスクの全領域の消去(フォーマット)、ファイル単位のバックアップ、リストアを行います。

① メインメニューから「ハードディスク・メンテナンス」にカーソル(反転表示)を合せて [↵] キーを押します。
② 表示が次のように変わります。実行する処理を選びます。

5.1　全領域・消去

ハードディスクを物理フォーマットします。

① サブメニューから「全領域・消去」を選び、[↵] キーを押します。
② フォーマットするハードディスクのドライブ番号を入力して [↵] キーを押します。
③ 20MBのハードディスクの場合は③-1が、40MBのハードディスクの場合は③-2の画面が表示されます。20MBの場合は標準フォーマットを行い、40MBの場合は拡張フォーマットを行います。20MBのハードディスクによっては拡張フォーマットできるものもありますが、ほかのアプリケーションソフトと分割して使用する場合は標準フォーマットを選びます。

③-1　　③-2

④　処理が終了すると表示がすべて［　　　］に変わります。

参考　ハードディスクを最初にフォーマットする場合に

正常にフォーマットされていないディスクです。

と表示されることがあります。これは異常ではありませんので、そのまま処理を続けてください。

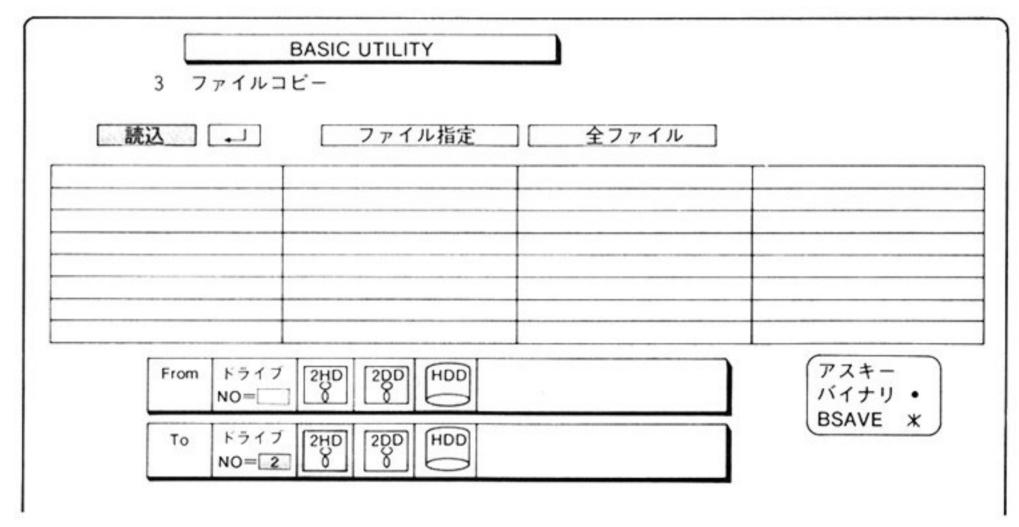
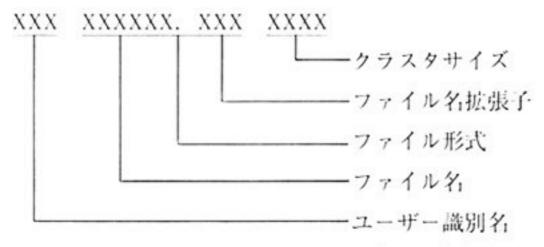
5.2　ファイルバックアップ

ハードディスクのBASIC領域中のファイルを1MBのフロッピーディスクにバックアップコピーします。バックアップの途中でフロッピーディスクがいっぱいになったときは、次のフロッピーディスクに続きをバックアップしてます。BASIC起動時のユーザー識別名の制限は受けず、すべてのファイルが処理の対象になります。「ファイルコピー」とはこの点が異なります。

フロッピーディスクはあらかじめ「フォーマット」の「データディスク作成」でフォーマットしてください。

①　サブメニューから「ファイルバックアップ」を選んで［↵］キーを押します。

②　バックアップを取りたいハードディスクのドライブ番号を入力します。

③　1MBフロッピーディスクをセットするドライブ番号を入力します。

④　「読込」が反転表示している状態で［↵］キーを押すとハードディスク内のすべてのBASICファイルが表示されます。

To側ドライブにはバックアップフロッピーの番号を"VOL NO."で表示します。

⑤　指定したファイルだけをコピーしたいときは「ファイル指定」を、ディスク上のすべてのファイルをコピーしたいときは「全コピー」を押します。

⑥ ファイル指定の場合はバックアップするファイルにカーソルを合せて [スペース] キーで選択します。

⑦ フロッピーディスクをセットします。

⑧ [f・5] キーを押すと反転表示したファイルを順次バックアップしていきます。コピーの終わったファイルは通常の表示に戻ります。

⑨ 途中でフロッピーディスクがいっぱいになったときは「*NO.XX シートをセットし [↵]」のメッセージが表示されます。フロッピーディスクを交換して [↵] キーを押します。

⑩ "VOL NO."が更新されたら [f・5] キーを押して処理を続行します。

⑪ 処理がすべて終了したら [ESC] キーを押してメニューに戻ります。

5.3 ファイルリストア

「ファイルバックアップ」でバックアップしたファイルをハードディスクに戻します。

① サブメニューから「ファイルリストア」を選んで [↵] キーを押します。

② バックアップしたディスクをセットするドライブ番号を入力します。

③ ハードディスクのドライブ番号を入力して [↵] キーを押します。

④ バックアップしたフロッピーディスクをセットします。

⑤ 「読込」が反転表示している状態で [↵] キーを押すとフロッピーディスク内のすべてのBASICファイルが表示されます。
From 側ドライブにはバックアップフロッピーの番号が"VOL NO."で表示されます。

⑥ 指定したファイルだけをリストアしたいときは「ファイル指定」を、すべてのファイルをリストアしたいときは「全コピー」を押します。

⑦ ファイル指定の場合はリストアするファイルにカーソルを合せて [スペース] キーで選択します。

⑧ [f・5] キーを押すと反転表示したファイルを順次リストアしていきます。リストアしたファイルは通常の表示に戻ります。

⑨ 複数のフロッピーディスクにまたがるファイルをリストアしたときは途中で「*次のシートをセットし [↵]」のメッセージが表示されます。この場合は次の番号のフロッピーディスクに交換して [↵] キーを押します。⑥の画面に戻りますので⑥以降の処理を繰返します。選択するファイルの先頭は前フロッピーでリストアが中断したファイルを指定してください。

6 メモリスイッチ変更

メモリスイッチの内容を機能ごと変更します。

① メインメニューから「メモリスイッチ変更」を選択し [↵] キーを押します。

② 表示が次のように変わり、カーソル移動キー [↓] [↑] で処理を選びます。

```
BASIC UTILITY

        M E N U
  1  ディスクバックアップ
  2  フォーマット
  3  ファイルコピー
  4  ハードディスク・メンテナンス
  5  メモリスイッチ変更
  6  日本語入力機能メンテナンス
  7  ユーザーフォント・メンテナンス
  8  拡張ハードコピー・オートスタート設定
  9  メニュー終了(BASIC に戻ります)

  1  RS232C(初期)
  2  RS232C(送受)
  3  メモリサイズ・他
  4  拡張ボード・他
  5  周辺装置
  6  拡張モニタ・他
```

③ それぞれのサブメニューの中では [↓] [↑] で機能を選び、[←] [→] で指定を選択します。メモリスイッチのシステム設定値と現在の設定値を画面右側に表示します。(使用する機種によってシステム設定値(既定値)が異なる場合があります。)

RS232C(初期)

BASIC UTILITY

5-1 メモリスイッチ変更 — RS232C(初期)

機能	指定	
Xパラメータ	**無効**	有効
通信方式	**全二重**	半二重
データビット長	**7ビット**	8ビット
パリティチェック	**なし**	あり
パリティ指定	**奇数**	偶数
ストップビット長	**1ビット**	1.5ビット
	2ビット	

	既定値	指定値
SW1	48	48
SW2	05	05
SW3	04	04
SW4	00	00
SW5	01	01
SW6	00	00

RS232C(送受)

BASIC UTILITY

5-2 メモリスイッチ変更 — RS232C(送受)

機能		指定				
ボーレート		75	150	130	600	**1200**
		2400	4800	9600		
日本語	KI=	**1B4B**		1A70		
シフトコード	KO=	**1B48**		1A71		
CR受信処理		**CR (0DH)**		CR・LF (0D0AH)		
リターンキー送信処理コード		**CR**		CR・LF		
Sパラメータ		**無効**		有効		

	既定値	指定値
SW1	48	48
SW2	05	05
SW3	04	04
SW4	00	00
SW5	01	01
SW6	00	00

メモリーサイズ・他

BASIC UTILITY

5-3 メモリスイッチ変更 — メモリサイズ・他

機能	指定				
メモリサイズ (Kバイト)	128	256	384	512	**640**
数値演算プロセッサ	**不使用**		使用		
テキスト画面の初期カラー	**白**		緑		
(ターミナル/入出力) DELコード受信動作	**BS/DEL**		NUL/NUL		

	既定値	指定値
SW1	48	48
SW2	05	05
SW3	04	04
SW4	00	00
SW5	01	01
SW6	00	00

拡張ボード・他

BASIC UTILITY

5-4　メモリスイッチ変更 — 拡張ボード・他

機　能	指　定	
サウンドボード	**不使用**	使　用
RS232C ボード	**不使用**	使　用
GPIB インターフェイスボード	**不使用**	使　用

	既定値	指定値
SW1	48	48
SW2	05	05
SW3	04	04
SW4	00	00
SW5	01	01
SW6	00	00

周辺装置

BASIC UTILITY

5-5　メモリスイッチ変更 — 周辺装置

機　能	指　定	
24ピン漢字 プリンタ	不使用	**使　用**
固定ディスク優先指定	**な　し**	あ　り
固定ディスクユーザ識別名	**使　用**	不使用
ハードコピー	**白　黒**	カラー
立ち上げ装置順位	**FDD→HDD**	2DD
	2HD	HDD♯1
	HDD♯2	

	既定値	指定値
SW1	48	48
SW2	05	05
SW3	04	04
SW4	00	00
SW5	01	01
SW6	00	00

拡張モニタ・他

BASIC UTILITY

5-6　メモリスイッチ変更 — 拡張モニタ・他

機　能	指　定	
モニタモード	**不使用**	使　用
画面ハードコピー拡張機能	**不使用**	使　用

	既定値	指定値
SW1	48	48
SW2	05	05
SW3	04	04
SW4	00	00
SW5	01	01
SW6	00	00

④　[f・5]キーを押すと、

＊ディップスイッチ2-5をONにし、
RESET後、変更されます。

と表示されます。

メモリスイッチの内容は電源を入れた直後に読み込まれます。したがってメモリスイッチの内容を変更した後は、リセットをするか、一度電源を切り再び電源を入れるようにしてください。

7　日本語入力機能メンテナンス

かな漢字変換方式に関する設定や、辞書のメンテナンスなどを行います。次の4つの処理を行うことができます。

(1)　環境設定

かな漢字変換機能を使用するかしないか、および、かな漢字変換機能を起動したときの動作環境などを設定します。設定できる項目は次のとおりです。

- かな漢字変換機能を使用するかしないか
- 郵便番号変換機能を使用するかしないか
- 日本語辞書ファイルのファイル名(ファイル名拡張子は .man .usr のみです。)
- 日本語辞書の辞書ドライブ
- BASICを起動したときの、かな漢字変換機能の入力モード
- BASICを起動したときの、かな漢字変換機能の変換モード
- 辞書学習を行うかどうか
- 改行(リターン)キーの機能
- スペースキーで変換を行うかどうか
- 郵便番号辞書の辞書ドライブ

設定した動作環境は、動作環境を登録したディスクでBASICを起動したときから有効になります。

必要なディスク

- 日本語Disk BASICユーティリティディスク
- 設定した環境を登録するディスク(日本語Disk BASICでフォーマット、システムプログラムの登録を行ったディスク)

(2)　日本語辞書

日本語辞書を編集します。このプログラムでは、次の3つの処理を行います。

- ユーザー辞書をメイン辞書に結合する

  ユーザー辞書には、最大で約4000語の単語を登録できます。4000語以上の単語を登録するには、現在ユーザー辞書に登録されている単語をメイン辞書に結合します。メイン辞書にユーザー辞書を結合するとユーザー辞書は空になりますので、さらに最大4000語の単語を登録できます。メイン辞書に結合した単語は削除できなくなりますので注意してください。

- ユーザー辞書を再編成してユーザー登録単語の登録場所を確保する

  ユーザー辞書には1000～4000語の単語を登録できますが、似たような読みの単語をいくつも登録すると、ユーザー辞書の中に空き領域があっても「ユーザー辞書の容量がいっぱいです」と

いうエラーメッセージを表示することがあります。このような場合、ユーザー辞書を再編成することによってユーザー単語を登録するための領域を確保します。

また、登録できる単語の数を変更することもできます。

- ユーザーが登録した単語の一覧を表示(印刷)する

ユーザーが登録した単語を確認するために、ユーザー登録単語を画面に表示したりプリンタに印刷したりします。

必要なディスク

- 日本語 Disk BASIC ユーティリティディスク
- 日本語 Disk BASIC のディスク

編集の対象となる日本語辞書(拡張子が man と usr のファイル)が登録されているもの。

注意 「日本語辞書」の「ユーザー辞書の結合」を実行する場合は、BASIC の起動時に表示される「How many files?(0-15)」というメッセージに対して5以上の値を、「ユーザー辞書の再編成」を実行する場合は3以上の値を、「ユーザー辞書の一覧出力」を実行する場合は2以上の値を設定してください。

「日本語辞書」の「ユーザー辞書の結合」「ユーザー辞書の再編成」を行う場合は、万一に備えて辞書ファイルのバックアップを取っておいてから実行してください。

「日本語辞書」の「ユーザー辞書の結合」はハードディスク上でしかできません。また、この場合、ハードディスク上の空き領域がメイン辞書(拡張子が man のファイル)の大きさの2倍以上必要になります。

(3) 郵便番号辞書

郵便番号ユーザー辞書を編集します。このプログラムでは、次の2つの処理を行います。

- 郵便番号で呼び出す住所を登録する。
- ユーザーが登録した住所を削除する。

必要なディスク

- 日本語 Disk BASIC ユーティリティディスク
- 日本語 Disk BASIC のシステムディスク

現在使用している郵便番号辞書(egzipm dic と egzipu dic)が登録されているもの。

注意 「郵便番号辞書」を実行する場合、郵便番号辞書が登録されているディスクは必ずドライブ1になければなりません。郵便番号辞書がハードディスクに登録されている場合は、郵便番号辞書をフロッピーディスクにコピーして、ドライブ1にセットして実行してください。

(4) ユーザー郵便番号辞書一覧表出力

ユーザーが登録した郵便番号と住所を表示します。

必要なディスク

- 日本語 Disk BASIC ユーティリティディスク
- 日本語 Disk BASIC のシステムディスク

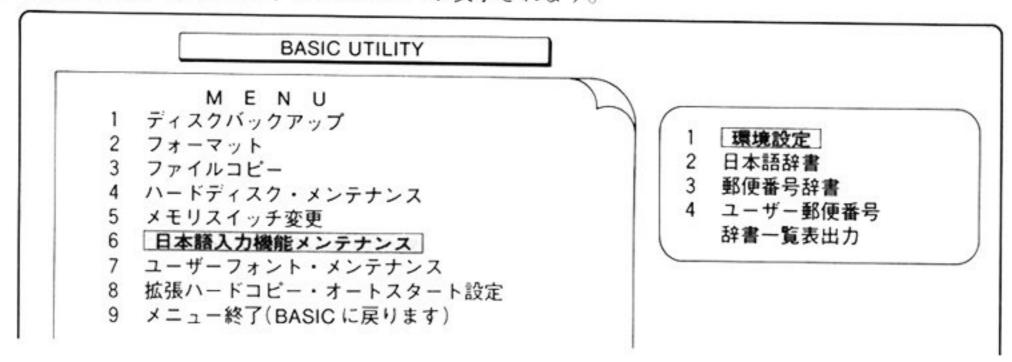
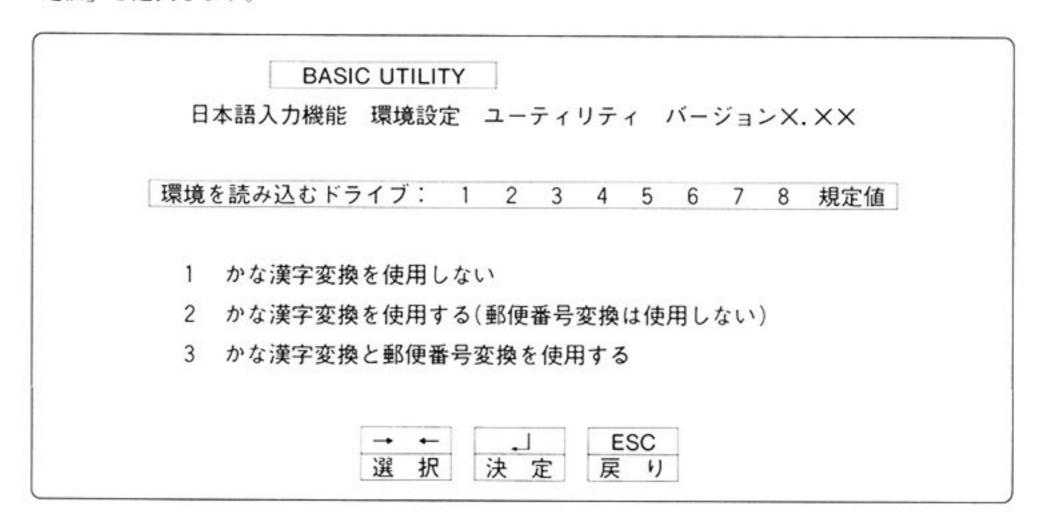
現在使用している郵便番号辞書(egzipm dic と egzipu dic)が登録されているもの。

日本語入力機能メンテナンスの起動

① メインメニューから「日本語入力機能メンテナンス」にカーソル(反転表示)を合わせて ⏎ キーを押します。

② 表示が次のように変わり、サブメニューが表示されます。

7.1 環境設定

① サブメニューから「環境設定」を選び ⏎ キーを押します。

② 表示が次のように変わります。日本語環境を読み込むディスクをセットするドライブのドライブ番号を選んで ⏎ キーを押します。日本語環境を初期設定の状態に設定する場合は「規定値」を選択します。

③ ↑ ↓ キーでかな漢字変換と郵便番号変換を使用するかどうかを選択し、⏎ キーを押します。

各項目の意味は次のとおりです。

- 「1　かな漢字変換を使用しない」

  かな漢字変換と郵便番号変換を使用しません。この場合の漢字の入力は、コード入力方式で行います。(「第4章　日本語入力」参照)

- 「2　かな漢字変換を使用する(郵便番号変換は使用しない)」

  かな漢字変換機能のみを使います。

- 「3　かな漢字変換と郵便番号変換を使用する」

  かな漢字変換機能と郵便番号変換機能を使います。

②で「規定値」を選択した場合は、初期状態として「3　かな漢字変換と郵便番号変換を使用する」が選択されています。

注意　「環境設定」で設定した値は、その設定を登録したディスクでBASICを立ち上げたときに有効になります。

7.1.1　かな漢字変換を使用しない場合の環境設定

①　「1　かな漢字変換を使用しない」を選び、［↵］キーを押します。

②　次のようなメッセージが表示されます。

環境設定するドライブ：1　2　3　4　5　6　7　8

設定した日本語環境を登録するディスクがセットされているドライブを選んで［f・5］キーを押します。

③　設定した環境が登録され、次のようなメッセージが表示されます。

※　設定した環境は、環境設定したディスクで起動すると、有効となります。

④　［ESC］キーを押してメインメニューに戻ります。

7.1.2　かな漢字変換だけを使用する場合の環境設定

①　「かな漢字変換を使用する（郵便番号変換は使用しない）」を選び、［↵］キーを押します。

②　表示が次のように変わります。［↑］［↓］キーで設定を行う項目を選び、［←］［→］キーでその項目に設定する動作条件を選びます。

日本語辞書ファイルのファイル名を変える場合は、［↑］［↓］キーで「辞書名（.man .usr）」にカーソルを合わせ、［DEL］キーまたは［BS］キーを押して表示されている辞書名を消してから新しい辞書ファイル名を入力します。辞書ファイル名は半角文字6字以内で、英数字またはカタカナでつけます。

```
                BASIC UTILITY
      日本語入力機能　環境設定　ユーティリティ　バージョンX.XX

辞書名( .man .usr)        ：egbdic
日本語辞書ドライブ：          1  2  3  4  5  6  7  8
入力モード：                ローマ字 ひらがな 全角    ローマ字 カタカナ 全角
                          ローマ字 カタカナ 半角    英 数    全角
                          英 数    半角
変換モード：                自由文入力変換           文章一括変換
辞書学習：                  す る                   しない
改行(リターン)キーの機能：確定し、次で改行            確定と同時に改行
スペースキーでの変換：        しない                  す る

          ↑ ↓     → ←     ↵      ESC
          機 能    選 択    決 定   戻 り
```

③　すべての項目を設定したら［↵］キーを押します。次のようなメッセージが表示されます。

環境設定するドライブ：1　2　3　4　5　6　7　8

設定した日本語環境を登録するディスクがセットされているドライブを選びます。

④ [f・5] キーを押すと、設定した日本語環境が登録されます。
この段階で日本語環境の設定をやり直したい場合は [ESC] キーを押します。[ESC] キーを1回押すとメッセージが消えて、環境を設定することができます。

⑤ 設定した日本語環境が登録され、次のようなメッセージが表示されます。

※ 設定した環境は、環境設定したディスクで起動すると、有効となります。

⑥ [ESC] キーを押してメインメニューに戻ります。

7.1.3 かな漢字変換と郵便番号変換を使用する場合の環境設定

① 「かな漢字変換と郵便番号変換を使用する」を選び、[↵] キーを押します。

② 表示が次のように変わります。[↑] [↓] キーで設定を行う項目を選び、[←] [→] キーでその項目に設定する動作条件を選びます。
日本語辞書ファイルのファイル名を変える場合は、[↑] [↓] キーで「辞書名(.man .usr)」にカーソルを合わせ、[DEL] キーまたは [BS] キーを押して表示されている辞書名を消してから新しい辞書ファイル名を入力します。辞書ファイル名は半角文字6字以内で、英数字またはカタカナでつけます。

```
                 BASIC UTILITY
      日本語入力機能　環境設定　ユーティリティ　バージョンX.XX

辞書名( .man .usr)        : egbdic
日本語辞書ドライブ:         1   2   3   4   5   6   7   8
入力モード:                ローマ字 ひらがな 全角   ローマ字 カタカナ 全角
                          ローマ字 カタカナ 半角   英 数    全角
                          英 数    半角
変換モード:                自由文入力変換           文章一括変換
辞書学習:                  す る                   しない
改行(リターン)キーの機能:  確定し、次で改行         確定と同時に改行
スペースキーでの変換:      しない                   す る
郵便番号辞書ドライブ:       1   2   3   4   5   6   7   8

          ↑ ↓    → ←    ↵      ESC
          機 能   選 択   決 定   戻 り
```

③ すべての項目を設定したら [↵] キーを押します。次のメッセージが表示されます。

環境設定するドライブ:1 2 3 4 5 6 7 8

設定した日本語環境を登録するドライブを選びます。

④ [f・5] キーを押すと、設定した日本語環境が登録されます。
日本語環境の設定をやり直す場合は [ESC] キーを押します。[ESC] キーを1回押すとメッセージが消えて、環境を設定し直すことができます。

⑤ 設定した日本語環境が登録されると次のメッセージが表示されます。

※ 設定した環境は、環境設定したディスクで起動すると、有効となります。

⑥ [ESC] キーを押してメインメニューに戻ります。

7.2 日本語辞書

注意 「日本語辞書」を実行するときは、「環境設定」で「かな漢字変換を使用しない」に設定したディスクで起動してください。
日本語辞書ファイルは書き込み禁止の属性にしないでください。書き込み禁止になっている場合はSETコマンドで書き込み禁止の属性を解除してください。
「ユーザー辞書の結合」を実行する場合は、BASICの起動時に表示される「How many files?(0-15)」というメッセージに対して5以上の値を、「ユーザー辞書の再編成」を実行する場合は3以上の値を、「ユーザー辞書の一覧出力」を実行する場合は2以上の値を設定してください。
「ユーザー辞書の結合」「ユーザー辞書の再編成」を行う場合は万一に備えて辞書ファイルのバックアップをとってから実行してください。
「メイン辞書とユーザー辞書の結合」はハードディスク上でしか行えません。また、この場合、ハードディスク上の空き領域がメイン辞書(拡張子がmanのファイル)の2倍以上必要になります。メイン辞書の大きさはFILESコマンドで、ハードディスク上の空き領域はDSKF関数で調べることができます。詳しくは「日本語Disk BASICリファレンスマニュアル」を参照してください。

① サブメニューから「日本語辞書」を選び、[↵]キーを押します。
② 表示が次のように変わります。
日本語辞書ファイルのファイル名を変える場合は、[DEL]キーまたは[BS]キーを押して表示されている辞書名を消してから新しい辞書ファイル名を入力します。辞書ファイル名は半角文字6字以内で、英数字またはカタカナでつけます。
[↓]キーを押してカーソルを「日本語辞書ドライブ」に移動して、日本語辞書が登録されているドライブ名を選んで[↵]キーを押します。

```
                BASIC UTILITY
     辞書メンテナンスユーティリティ　バージョンX.XX

辞書名( .man .usr)      ：egbdic
日本語辞書ドライブ      ：1  2  3  4  5  6  7  8

               ↑ ↓    → ←    ┘      ESC
               機 能   選 択   決 定   戻 り
```

③　表示が次のように変わります。

```
          BASIC UTILITY
辞書メンテナンスユーティリティ　バージョンX.XX

          ユーザー辞書の結合
          ユーザー辞書の再編成
          ユーザー辞書の一覧出力
          終　　了

カーソルで機能を選択してリターンキーを押してください。
```

各項目の意味は次のとおりです。

- **「ユーザー辞書の結合」**
  ユーザー辞書に登録した単語をメイン辞書に登録します。メイン辞書へ登録された単語はユーザー辞書から削除されます。
- **「ユーザー辞書の再編成」**
  ユーザー辞書を再編成してユーザー辞書に単語を登録する領域を確保します。また、ユーザー辞書に登録できる単語の最大数を設定します。
- **「ユーザー辞書の一覧出力」**
  ユーザー辞書に登録した単語を画面に表示したりプリンタに印刷したりします。

7.2.1　ユーザー辞書の結合

ユーザー辞書の結合は辞書ファイルがハードディスクに入っている場合のみ実行できます。

注意　ユーザー辞書の結合を行う場合は、万一に備えて辞書ファイル(拡張子がmanのファイルとusrのファイル)はフロッピーディスクなどにバックアップしてください。バックアップの方法については「2　ディスクバックアップ」または「4　ファイルコピー」の項を参照してください。

①　「ユーザー辞書の結合」を選び、[↵]キーを押します。

②　表示が次のように変わります。結合するメイン辞書(拡張子がmanのファイル)とユーザー辞書(拡張子がusrのファイル)が登録されているハードディスクのドライブ番号を入力して、「実行」にカーソルを合わせて[↵]キーを押します。

BASIC UTILITY

辞書メンテナンスユーティリティ　バージョンX.XX

＊＊＊＊　ユーザー辞書の結合　＊＊＊＊

辞書をドライブ 3 ：に入れてください。

実　行　　　　　中　止

③ 次のメッセージが表示され、メイン辞書とユーザー辞書の結合が実行されます。

作業終了まで最大30分必要です。

④ ユーザー辞書の結合が実行されている間は**「ユーザー辞書の結合中です。しばらくお待ちください。」**のメッセージがゆっくりと点滅します。

ユーザー辞書中に同音語(同じ読みの単語)が数多くある場合、その単語はメイン辞書に結合できないことがあります。この場合ピッというブザーが鳴り、表示が次のように変わってメイン辞書に登録できなかった単語と単語の読みを表示します。

任意のキーを押すと表示が元に戻り、ユーザー辞書の結合を続行します。

注意　ここで表示された単語はメイン辞書に結合されません。このためユーザー辞書の結合が終了した後もその単語はユーザー辞書に残ったままになっています。

BASIC UTILITY

辞書メンテナンスユーティリティ　バージョンX.XX

＊＊＊＊　ユーザー辞書の結合　＊＊＊＊

辞書をドライブ 3 ：に入れてください。

作業終了まで最大30分必要です。

同意語が多すぎます。　　　読み　　単語

何かキーを押すと処理を続行します。

⑤ メイン辞書とユーザー辞書の結合が終わると、次のメッセージが表示されます。

ユーザー辞書の結合を終わりました。何かキーを押してください。

⑥ 任意のキーを押すと、ユーザー辞書の結合を終了して、日本語辞書メンテナンスのメニュー画面に戻ります。

7.2.2 ユーザー辞書の再編成

注意 ユーザー辞書の再編成を行う前に、万一に備えて辞書ファイル(拡張子がmanのファイルとusrのファイル)は別のフロッピーディスクなどにバックアップしてください。バックアップの方法については「2　ディスクバックアップ」または「4　ファイルコピー」を参照してください。

① 「ユーザー辞書の再編成」を選び、[↵]キーを押します。

② 表示が次のように変わります。再編成を行うユーザー辞書が登録されているハードディスクのドライブ番号を入力して、[↵]キーを押します。

BASIC UTILITY

辞書メンテナンスユーティリティ　バージョン×.××

＊＊＊＊　ユーザー辞書の再編成　＊＊＊＊

辞書をドライブ[3]：に入れてください。

実　行　　　　　中　止

③ 次のメッセージが表示されます。ユーザー辞書の再編成を行った後に、ユーザー辞書に登録できる単語の数を設定します。登録数は、1000、2000、3000、4000の4種類の中から選びます。1～4の数字を入力して選んでください。初めに表示されるのは現在の登録数です。

再編成後に登録できる最大数は、約×000語です。

④ 「実行」にカーソルを合わせ、[↵]キーを押すとユーザー辞書の再編成が実行され、次のメッセージが表示されます。

作業終了まで最大30分必要です。

ユーザー辞書の結合が実行されている間は「**ユーザー辞書の再編成中です。しばらくお待ちください。**」のメッセージがゆっくりと点滅します。

⑤ ユーザー辞書の再編成が終わると、次のメッセージが表示されます。

ユーザー辞書の再編成を終わりました。何かキーを押してください。

⑥ 任意のキーを押すと、ユーザー辞書の再編成を終了して、日本語辞書メンテナンスのメニュー画面に戻ります。

7.2.3 ユーザー辞書の一覧出力

① 「ユーザー辞書の一覧出力」を選び、 ⏎ キーを押します。

② 表示が次のように変わりますので、指示されているドライブに日本語辞書を登録したディスクをセットします。表示されているドライブとは別のドライブに日本語辞書を登録したディスクをセットしたい場合は、数字キーで表示されているドライブ番号を変更します。

```
              BASIC UTILITY
     辞書メンテナンスユーティリティ  バージョンX.XX

     ＊＊＊＊  ユーザー辞書の一覧出力  ＊＊＊＊

辞書をドライブ 3 ：に入れてください。



    実 行          中 止
```

③ 「実行」にカーソルを合わせて ⏎ キーを押します。

④ 表示が次のように変わりますので、一覧出力をする単語の読みの範囲を指定します。
まず出力を開始する単語の読みを入力します。五十音順に並べたときに、ここで入力した読みよりも前になる読みのユーザー登録単語は出力しません。
初期設定は「あ」になっています。別の読みを入力する場合は、 BS キーを押して「あ」を消してから、読みを入力します。

```
              BASIC UTILITY
     辞書メンテナンスユーティリティ  バージョンX.XX

     ＊＊＊＊  ユーザー辞書の一覧出力  ＊＊＊＊

読みの開始：あ■
読みの終了：

出力先：
    ■画面
    □プリンタ
```

⑤ 出力を開始する単語の読みを入力したら［↵］キーを押します。

⑥ カーソルが移動して、表示が次のように変わりますので、出力を終了する単語の読みを入力します。五十音順に並べたときに、ここで入力した読みよりも後ろになる読みのユーザー登録単語は出力しません。
初期設定は「ん」になっています。別の読みを入力する場合は、［BS］キーを押して「ん」を消してから、読みを入力します。

⑦ 出力先を選択します。「プリンタ」を選ぶときは、プリンタが接続されていることを確認してください。
「実行」にカーソルを合わせ、［↵］キーを押すと一覧出力を開始します。

BASIC UTILITY

辞書メンテナンスユーティリティ　バージョンX.XX

****　ユーザー辞書の一覧出力　****

読みの開始：あ

読みの終了：ん

出力先：

■画面

□プリンタ

実　行　　　　中　止

⑧ ユーザー登録単語の出力先に「画面」を選んだ場合、登録単語が読みの五十音順に表示されます。一つの画面に10個の登録単語が表示されます。登録単語が11語以上ある場合は、次のようにして表示を切り換えます。

- 「次ページ」にカーソルを合わせて [↵] キーを押す、または [↓] キーを押す。
  表示されている単語の次の10語を表示します。最後の登録単語を表示する画面が表示されたときに「次ページ」を行うと「これより先のページはありません。」と表示されます。
- 「前ページ」にカーソルを合わせて [↵] キーを押す、または [↑] キーを押す。
  直前に表示していた10語を表示します。最初の登録単語を表示する画面が表示されたときに「前ページ」を行うと「これより前のページはありません。」と表示されます。

「終了」にカーソルを合わせて [↵] キーを押します。
ユーザー登録単語の一覧表示を終了します。

ユーザー登録単語の出力先に「プリンタ」を選んだ場合、次のメッセージがゆっくりと点滅し、ユーザー登録単語を読みの五十音順にプリンタに印刷します。１ページに50個の登録単語が印刷されます。

プリンタに出力中です。ESC キーを押すと中断します。

プリンタへの出力が終わると、自動的に日本語辞書メンテナンスのメニュー画面に戻ります。
また、プリンタへの出力を途中で中止する場合は [ESC] キーを押します。[ESC] キーを押すと、自動的に日本語辞書メンテナンスのメニュー画面に戻ります。

- ユーザー辞書に単語が登録されていない場合は**「登録されていません。」**というメッセージが表示されて、自動的に日本語辞書メンテナスのメニュー画面に戻ります。

7.3 郵便番号辞書

① サブメニューから「郵便番号辞書」を選んで、[↵] キーを押します。
② 表示が次のように変わります。

ドライブ１にシステムディスクを入れてください。

[実行]

ドライブ１に郵便番号辞書(egzipm. dic と egzipu. dic)が登録されているシステムディスクを挿入し、[↵] キーを押します。

③ 住所を登録または削除する郵便番号を入力して [↵] を押します。
指定した郵便番号に登録されている住所を表示しますので、次のキーを使って処理を選びます。

キー	説明
[↑] [↓]	指定した郵便番号にユーザーが登録した住所が２つ以上あるときに、住所を登録する場所、または削除する住所を選びます。
[f・1]	住所の登録または削除を終了します。
[f・2]	カーソルのある位置に表示されている住所を郵便番号ユーザー辞書から削除します。
[f・3]	カーソルのある位置に住所を登録します。

```
＊＊＊　郵便番号ユーザー辞書登録プログラム　＊＊＊

郵便番号＝３９９

1　＊長野県松本市
2　＊長野県塩尻市
3
4
5
6
7
8
9

終了・削除・追加　　どれを実行しますか　　(f・1：終了　f・2：削除　f・3：追加)
```

注意　住所の左に「＊」が表示されているものは、郵便番号メイン辞書に登録されている住所ですので、削除できません。

7.3.1 住所の登録

① ユーザーが登録した住所が2つ以上ある場合、カーソル移動キー［↑］［↓］を使って住所を登録する場所を選びます。

② ［f・3］キーを押します。

③ 登録する住所を入力します。住所の長さは、全角文字で16文字分の長さまでです。

④ 住所を入力したら［↵］キーを押します。

7.3.2 住所の登録・削除の終了

① ［f・1］を押すと、住所の登録・削除の処理を終了します。

② 編集を行う郵便番号を入力する画面に戻ります。続けて編集を行うときは新たな郵便番号を入力します。郵便番号ユーザー辞書の編集を終わるときは［f・1］を押します。

③ BMENUをドライブ1から起動したときは、ドライブ1にユーティリティディスクを挿入し、［↵］を押して、メニュー画面にもどります。

7.4 ユーザー郵便番号辞書一覧表出力

① サブメニューから「ユーザー郵便番号辞書一覧表出力」を選んで、［↵］キーを押します。

② 表示が次のように変ります。

```
ドライブ1にシステムディスクを入れてください。

［実行］
```

③　ドライブ1にegzipu　dicが登録されているシステムディスクを挿入して［↵］キーを押します。

④　郵便番号と住所を表示し終わると「どれかキーを押してください。メニューに戻ります。」と表示しますので、何かキーを押してください。

⑤　ドライブ1にユーティリティディスクを挿入して［↵］キーを押します。
BMENUのメニュー画面に戻ります。

8　ユーザーフォント・メンテナンス

ユーザーが定義する文字フォント(外字)の登録または更新のための機能です。ここで作成したファイルをuserfo.ntというファイル名で登録すればシステム起動時に自動的にこのファイルを読み込み、このファイルに登録されている外字を使用することができます。またこのファイル名を変更することもできます。外字を使用するには登録した16進コードを使用します。

①　メインメニューから「ユーザーフォント・メンテナンス」を選び、［↵］キーを押します。
表示が次のように変わります。

更新または登録ですか(U/L)

外字ファイルの修正・追加を行う場合はU(更新)を、またuserfo.nt以外のファイルを外字ファイルとして使用する場合はL(登録)を入力します。以下、更新と登録に分けて説明します。
処理を中止する場合は、［ESC］キーを押してBMENUのメニュー画面に戻ります。

8.1　更新

新たに文字フォントを作ったり、既に登録してあるフォントを変更する場合、あるいは標準ファイル名(userfo.nt)以外の外字ファイルを作ります。

①　Uを入力すると

入力ファイル名［1：userfo.nt］：

と表示します。userfo.ntは外字ファイルの標準ファイル名です。別のファイル名の外字ファイルの内容を変更する場合はその名前を入力し、そうでなければ［↵］キーだけを押します。

②　次のメッセージを表示します。

出力ファイル名［1：userfo.nt］：

作った文字フォントを登録するファイル名を入力します。更新の場合は入力ファイル名と同じファイル名を入力します。新たにファイルを作る場合は別のファイル名を入力します。標準ファイルの場合は［↵］キーだけを押します。

③　次のメッセージを表示します。

始めますか(Y/N)?

ファイル名を間違えたときなどに「N」を入力すると最初のメッセージに戻ります。「Y」を入力するとフォント編集画面を表示します。フォント編集画面にはコマンドモードとエディットモードがあり、［↵］キーの入力でモードが切り替わります。

```
                ユーザー定義外字文字保守        Ver X.X

L：登録    カーソルキー：フォント選択                1～9：カーソル移動
E：終了              リターンキーで決定              ＊   ：ドットオン
C：クリア         G：コード呼び出し                  ＋   ：ドットオフ
R：反転                                              ＝   ：オンオフ取り消し



            0 1 2 3 4 5 6 7 8 9 A B C D E F
7620          □
7630                                                        7621
7640
7650                                                    ┌──────────┐
7660                                                    │          │
7670                                                    │ フォント  │
7720                                                    │          │
7730                                                    │ 作成領域  │
7740                                                    │          │
7750                                                    └──────────┘
7760
7770
```

コマンドモード

L　入力ファイルの登録されている外字および修正または追加した外字をシステムに登録します。この段階では外字用メモリに登録するだけなので、システムをリセットするか電源を切るまでしか登録した外字は使用できません。終了処理(E)でファイルを更新して初めて出力ファイルに登録します。

E　処理を終了します。次のメッセージは表示します。

外字用メモリに登録しますか(Y/N)?

「Y」を入力すると外字用メモリに登録します。登録するとリセットするか電源を切るまで登録した外字を使用できます。「N」を入力すると外字用メモリに登録しません。どちらの場合も次のメッセージを表示します。

ファイルを更新しますか(Y/N)?

「Y」を入力すると変更したり修正した文字フォントを出力ファイルに書き込み、ユーザーフォントメンテナンスを終了します。「N」を入力すると出力ファイルを更新せずに終了します。

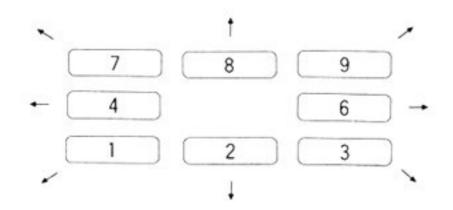
カーソルキー　外字選択のカーソル □ を移動します。カーソルが位置しているコードに対する文字フォントを作ったり修正することができます。画面の右のフォント作成領域の上に文字コードを表示します。

[↵] エディットモードに入ります。フォント作成領域の左上にエディットモードのカーソルを表示します。

エディットモード

C　現在フォント作成領域にあるフォントを消去(クリア)します。

R　現在フォント作成領域にあるフォントを反転表示します。

数字キー　カーソルを移動します。

*　カーソル位置のドットをオン(表示)します。カーソルは+の反転表示に変わります。この状態でカーソルを移動すると移動先のドットを表示します。この状態は [+] または [=] を押すまで続きます。

+　カーソル位置のドットをオフ(消去)します。カーソルは+に変わります。この状態でカーソルを移動すると移動先のドットを消去します。この状態は [*] または [=] を押すまで続きます。

=　カーソルを=にします。この状態でカーソルを移動しても、ドット状態は変更しません。

G　すでに登録済みの外字や漢字 ROM に記憶している文字フォントをフォント作成領域に取り込みます。[G] を押すと

呼び出す漢字の番号を入力してください。

と表示します。呼び出す漢字の漢字コード(JIS コード)を入力します。例えば「〒」は 2229 です。[↵] キーを押すと、そのコードを含む同じページのフォントをすべて表示します。

入力したコードの位置にカーソルがあります。カーソル移動キーでコードを変更できます。[ESC] キーを押すとコード入力の状態に戻りますので、別のパターンを表示しなおすことができます。もう一度 [↵] キーを押すとフォントがフォント作成領域に取り込まれ既にフォント作成領域にあるフォントと合成します。既存のフォントを利用して新しいフォントを作るのに便利な機能です。フォント取り込み後はエディットモードになります。

[↵] フォント作成領域のフォントを確定し、コマンドモードに戻ります。カーソルは次のコードに移動します。

ESC　現在作成中のフォントの編集を中止して、コマンドモードに戻ります。カーソルは次のコードに移動します。

8.2 登録

外字ファイルの標準ファイル名は userfo.nt です。BASIC 起動時にはこのファイルを読み込み、外字コードを入力するとこのファイルに登録してあるフォントが使用されます。この名前以外の外字ファイルを使用したいときは「L」を入力します。次のメッセージを表示します。

入力ファイル名［1：userfo.nt］：

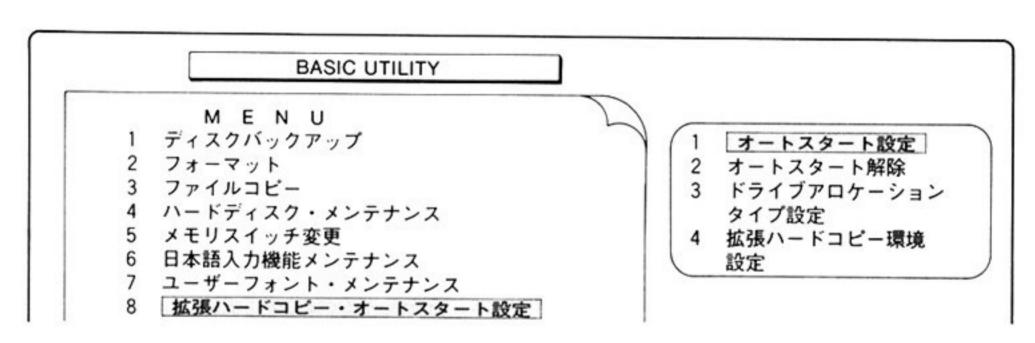
使用する外字ファイルのファイル名を入力します。次のメッセージを表示し実行を終了します。

登録が終了しました。

変更後はここで指定した外字ファイルを読み込みます。

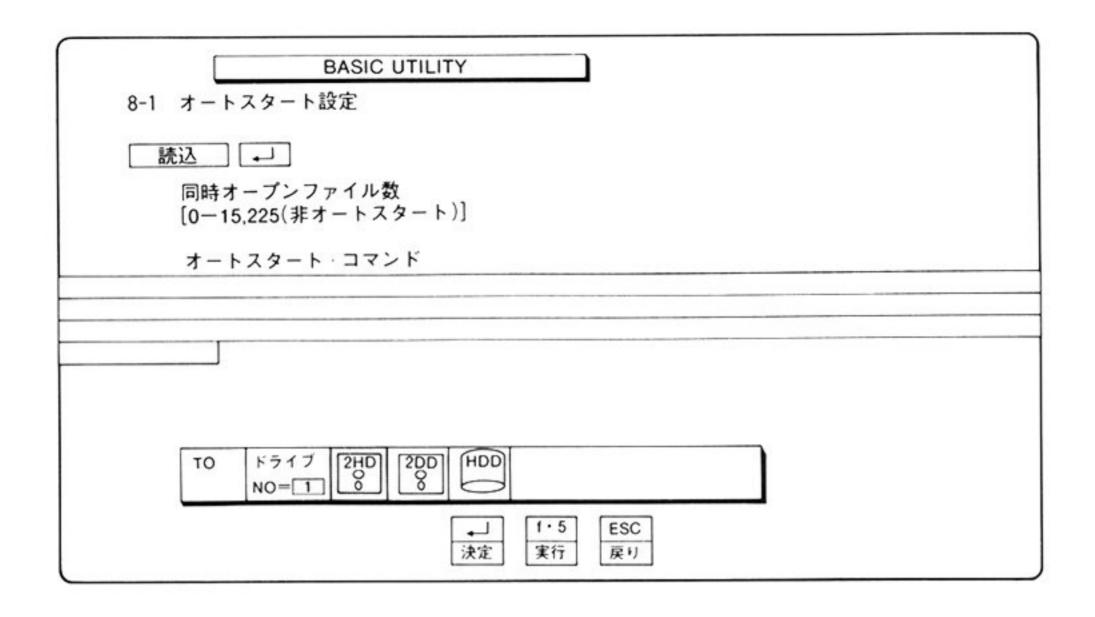
9 拡張ハードコピー・オートスタート設定

システムのオートスタートの設定・解除およびフロッピーディスクドライブのアロケーションタイプを変更します。

① メインメニューから「拡張ハードコピー・オートスタート設定」を選び、⏎キーを押します。表示が次のように変わります。

② サブメニューで、項目を選び、⏎キーを押します。

9.1 オートスタート設定

BASIC が起動した直後に、自動的に BASIC のプログラムを実行する機能です。

① サブメニューから「オートスタート設定」を選び、⏎キーを押します。
② オートスタート設定用のシステムディスクをセットするドライブ番号とディスクの種類を選び⏎キーを押します。
③ システムディスクを指定したドライブにセットし⏎キーを押します。次の画面を表示します。

④ BASIC 起動時に How many files?(0−15)に答えるファイルの数を入力し [↵] キーを押します。入力できる数は 0 から15までの整数です。また255を入力するとオートスタートの解除になります。

⑤ 続いてオートスタートコマンドを入力します。最大253文字まで入力できます。ここで入力したコマンドを BASIC 起動時に実行します。

⑥ [f・5] キーを押すと処理が終了します。

9.2 オートスタートの解除

オートスタートを解除します。

① サブメニューから「オートスタート解除」を選び、[↵] キーを押します。

② オートスタートの設定を解除するシステムディスクをセットするドライブ番号を入力し、カーソル移動キー [←] [→] でディスクの種類を選びます。

③ [↵] キーを押します。

④ システムディスクを指定したドライブにセットし、[f・5] キーを押します。

⑤ [ESC] キーを押します。

9.3 ドライブアロケーションタイプ設定

システムディスク上のドライブアロケーションテーブルの設定を変える機能です。これによりドライブの優先順位が変わります。

① サブメニューから「ドライブアロケーションタイプ設定」を選び、[↵] キーを押します。

② ドライブアロケーションテーブルを変えるシステムディスクをセットするドライブのドライブ番号とディスクの種類を選び、[↵] キーを押します。

③ システムディスクを指定したドライブにセットし、[↵] キーを押します。ディスクの内容を読み込み、現在の設定を表示します。

BASIC UTILITY

8-3　ドライブアロケーションタイプ変更

読込　↵

	タイプ2	タイプ1
順位	1MB ↓ 640K	640K ↓ 1MB

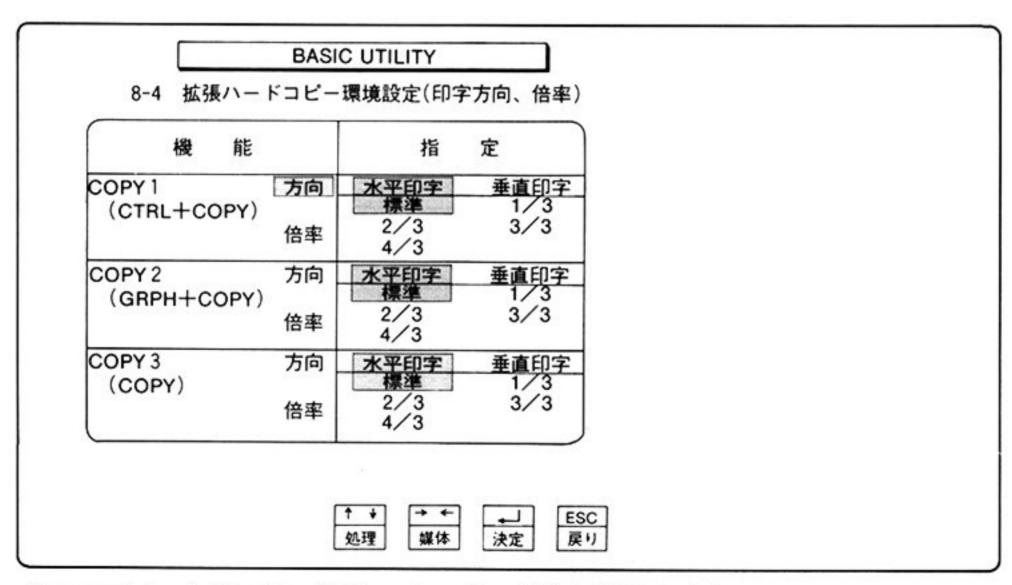
④　カーソル移動キー［←］［→］でアロケーションタイプを選択します。

⑤　［f・5］キーを押します。ドライブアロケーションを変更し、**処理終了**のメッセージを表示します。

9.4　拡張ハードコピー環境設定

ページプリンタを使用しているときに、ページプリンタの機能を生かしたハードコピーを実行することができます。ページプリンタ以外でこの機能を使っても正常動作しませんので注意してください。

①　サブメニューから「ハードコピー環境設定」を選んで［↵］キーを押します。

②　カーソル移動キー［→］［←］［↑］［↓］を使って COPY 1 コマンド、COPY 2 コマンド、COPY 3 コマンドで実行される機能を設定します。ページプリンタ以外の場合はすべて標準に設定してください。

③　［f・5］キーを押すと、拡張ハードコピーの機能が設定されます。
新しく BASIC を起動したときから、この設定が使用できます。

注意　ページプリンタは PC-PR601 系と同等の機能を持ったページプリンタのことです。

10　通信ユーティリティ

日本語 Disk BASIC の動作モードには、BASIC のコマンド等を実行する BASIC モードと、データ通信を行うターミナル・モードがあります。ターミナル・モードではコンピュータの RS-232C インターフェイスを介して接続されている機器とデータ通信を行います。

通信ユーティリティ「BTERM」では、BASIC モードからターミナル・モードに制御を移してデータ通信を行います。

またこのほかに、通信パラメータの設定、通信内容の表示や印刷、ファイルの編集やアップロード、ダウンロード、などを行うことができます。

(1)　ターミナル・モード

BASIC をターミナル・モードにして、RS-232C インターフェイスを介して接続している機器とデータ通信を行います。送受信したデータはスクロールバッファに記憶され、同時に画面にも表示されます。

送信するデータは次の 2 つのいずれかのモードで入力します。

- 1 文字入力モード：英数字とカタカナのみ入力できます。
- 1 行入力モード　：日本語入力機能を使って、漢字の入力ができます。

通常の場合は 1 文字入力モードでデータを入力し、漢字を入力するときだけ 1 行入力モードにします。

注意　通信ユーティリティ「BTERM」は漢字コードとして JIS コードを採用しています。したがって漢字コードとしてシフト JIS コードを採用している相手と通信することはできません。特に一般のパソコン通信サービスにはシフト JIS コードを採用しているものもありますので注意してください。

ターミナル・モードでは画面が次のようになります。

パラメータ設定	ダウンロード	アップロード	プリント	終　了

またターミナル・モードでは次の処理を行うことができます。

- ダウンロード
  ダウンロードとは、データの通信内容をファイルに書き込むことです。
  次の場合、ダウンロードを自動的に中止します。
  ・ダウンロード用ファイルの大きさが400行を越えた場合。
  ・ディスクの空き領域がなくなった場合。
  ダウンロードしたファイルは通信ユーティリティの「エディタ・モード」で編集することができます。
- アップロード
  アップロードとは、ファイルの内容を通信相手に送信することです。アップロードするファイルは通信ユーティリティの「エディタ・モード」で作成、編集します。
- スクロール・バッファの参照
  ターミナル・モードでの通信の内容はメモリ上のスクロール・バッファと呼ばれる場所に記憶されます。通信ユーティリティでは、スクロール・バッファに記憶された通信内容を表示することができます。スクロール・バッファの大きさは約400行分です。通信内容の大きさがこれよりも大きくなった場合は、古いものから順番に通信内容を削除します。
- 印刷
  通信の内容をプリンタで印刷します。
- 通信パラメータの設定
  各種の通信パラメータを設定します。ここで設定した通信パラメータの値はターミナル・モード中のみ有効です。ターミナル・モードを終了すると、自動的に「(3)通信パラメータ設定」で設定された値に戻ります。

(2) エディタ・モード

アップロードするファイルの編集や作成、ダウンロードしたファイルの編集などを行います。またファイルの内容を印刷することもできます。

アップロード、ダウンロード用のファイルの大きさは最大400行です。なお、この値は1行中の文字数には関係ありません。ファイル中に空白の行がある場合も、その行を1行として数えます。

表示領域(22行)

入力行(1行)

文頭へ　文末へ　行挿入　行削除　行複写　行貼込　文削除　文読込　印　刷　終　了

ファイルの編集のために編集モードと参照モードの２つのモードがあります。編集したい行を選ぶときは参照モード、編集するときは編集モードにします。

(3) 通信パラメータ設定

データ通信の各種パラメータを設定します。設定された通信パラメータの値はパラメータファイルに書き込まれ保存されます。設定した値は、「通信パラメータ設定」を終了した時点で有効になります。

通信ユーティリティ　パラメータ設定

機　能	指　定	
パリティチェック	なし	奇数
	偶数	
データビット長	7ビット	8ビット
ストップビット長	1ビット	1.5ビット
	2ビット	
Xパラメータ	無効	有効
Sパラメータ	無効	有効
DELコード受信動作	BS	NUL
リターンキー送信処理コード	CR	CR LF

コマンド入力	TERM"N81XNBCLP", F

CR受信処理	CR	CR LF
日本語シフトコード KI= KO=	1B4B 1B48	1A70 1A71
通信方式	全二重	半二重

機　能	指　定		
ボーレート	300	1200	2400
	4800	9600	

注意　「ターミナル・モード」または「通信パラメータ設定」で設定した通信パラメータの値は、メモリスイッチの値とは関係ありません。ここで設定した値は、通信ユーティリティでの通信時のみ有効です。

「通信パラメータ」で通信パラメータの値を設定した場合、パラメータファイルに書き込まれた値は再度「通信パラメータ設定」で設定し直さない限り変わりません。

「ターミナル・モード」で設定した通信パラメータの値はパラメータファイルには書き込まれません。したがってターミナル・モードを終了して、次回にターミナル・モードを起動したときの通信パラメータの値は、「通信パラメータ設定」で設定した値に戻ります。

10.1 BTERMの起動と終了

BTERMを実行するには、ディスク上に次のファイルが必要です。

BTERM.	BTERM. EDT
BTERM. SET	BTERM. TRM

10.1.1 BTERMの起動

① 日本語Disk BASICユーティリティディスクをドライブ1にセットし、次のように入力します。

RUN"BTERM [⏎]

② 次のメニューが表示されます。
実行する処理を [↑] [↓] キーあるいは数字キーを使って選び、[⏎] キーを押します。

通信ユーティリティー

MENU

1 ターミナル・モード
2 エディタ・モード
3 通信パラメータ設定

9 終 了

③ 選択した処理の画面が表示されます。画面の下部や、表示されるメニューの中に、その処理での各キーの使用方法を表示しますのでその指示に従ってください。

各処理に共通なキー操作
処理の実行 ： [f・5] キー
処理の決定 ： [⏎] キー
前の画面に戻る ： [ESC] キー
項目、機能の選択： [↑] [↓] [←] [↑] キー

10.1.2 BTERMの終了

① 各処理を終了して、起動時のメニュー画面に戻ります。
② メニュー画面で「9 終了」を選び [⏎] キーを押すと通信ユーティリティの実行を終了し、入力モードになります。

10.2 ターミナルモード

① メニュー画面から「1　ターミナル・モード」を選び、[↵]キーを押します。
② BASICモードからターミナルモードに変わり、表示が次のようになります。この状態で、RS-232Cインターフェイスを介して接続されている機器との通信を行います。

パラメータ設定	ダウンロード	アップロード	プリント	終　了

ターミナルモードではデータの送受信を行うほかにもいろいろな機能が実行できます。これらの機能は[f・1]～[f・5]のファンクションキーで選びます。画面の下側にはそれぞれのキーの機能が表示されています。

各キーの機能は次のとおりです。

[f・1]キー(パラメータ設定)：通信パラメータを設定します。
[f・2]キー(ダウンロード)　：データのダウンロードを開始、終了します。
[f・3]キー(アップロード)　：データのアップロードを開始、終了します。
[f・4]キー(プリント)　　　：通信内容をプリンタに印刷します。
[f・5]キー(終了)　　　　　：ターミナルモードを終了します。
[f・10]キー　　　　　　　：入力モードを切り換えます。
[HELP]キー　　　　　　　 ：スクロールバッファ参照モードにします。

注意　通信ユーティリティ「BTERM」は漢字コードとしてJISコードを採用しています。したがって漢字コードとしてシフトJISコードを採用している相手と通信することはできません。特に一般のパソコン通信サービスにはシフトJISコードを採用しているものもありますので注意してください。

10.2.1 データの入力

通信相手に送信するデータの入力方法には「一文字入力モード」と「一行入力モード」の二つがあります。この二つの方法は[f・10]キーを押して切り換えます。

• 一文字入力モード

ターミナルモードを起動したときの入力モードです。

入力できる文字は英数字とカタカナのみで、データの送信はキーボードから一文字入力するごとに

行われます。

画面にはキーボードから入力した内容と、受信した通信内容が表示されます。

- 一行入力モード

日本語入力機能を使って漢字を入力することができます。また、データの送信は行単位で行われ、[↵]キーを押すまで送信されません。

一行入力モード中は、画面にはキーボードから入力した内容のみ表示され、受信した通信内容は表示されません。一行入力モード中に受信したデータは、一行入力モードを終了して一文字入力モードに戻ると表示されます。

一行入力モードでは、日本語入力機能を使って漢字を入力することができますが、一行入力モードにしている間は通信相手から受信した内容はスクロール・バッファに記憶されるだけで表示されません。通常の場合は「一文字入力モード」で送信内容を入力し、漢字を入力するときだけ「一行入力モード」に切り換えて入力するようにしてください。

注意 「通信パラメータ設定」で、通信方式を「全二重」に設定した場合、入力した内容は表示されません。

一文字入力モードと一行入力モードの切り換え方法は次のとおりです。

① 漢字を入力する必要ができたら[f・10]キーを押して一行入力モードにします。

② 漢字の入力が終わったら[f・10]キーを押して一文字入力モードに戻ります。

一行入力モード時は、次の点に注意してください。

- [↵]キーを押すまで入力したデータは送信されません。
- 一行入力モード中に受信データがあった場合、一文字入力モードに戻るまでそのデータは表示されません。
- 漢字の入力は[CTRL]+[XFER]キーを押して日本語入力機能を起動して行います。漢字の入力の方法については「第4章　日本語入力」を参照してください。

10.2.2 パラメータ設定

注意 通信パラメータの設定にはターミナル・モードで通信中に設定する方法と、通信ユーティリティのメニュー画面の「通信パラメータ設定」で設定する方法とがあります。
ターミナル・モードで通信パラメータを設定した場合、ターミナル・モードを終了するとその設定は消去されます。

① [f・1]キーを押します。

② 表示が次のように変わります。この画面で通信パラメータを設定します。通信パラメータの設定方法については「10.1　通信パラメータ設定」を参照してください。

通信ユーティリティ　パラメータ設定

機　能	指　定	
パリティチェック	なし	奇数
	偶数	
データビット長	7ビット	8ビット
ストップビット長	1ビット	1.5ビット
	2ビット	
Xパラメータ	無効	有効
Sパラメータ	無効	有効
DELコード受信動作	BS	NUL
リターンキー送信処理コード	CR	CR LF
コマンド入力	TERM"N81XNBCLP", F	

CR受信処理	CR	CR LF
日本語シフトコード KI=	1B4B	1A70
日本語シフトコード KO=	1B48	1A71
通信方式	全二重	半二重

機　能	指　定		
ボーレート	300	1200	2400
	4800	9600	

③ 通信パラメータを設定したら f・5 キーを押してください。パラメータ設定を終了し、ターミナル・モードに戻ります。これ以降、ターミナル・モードを終了するまで通信パラメータの値は②で設定した値になります。

10.2.3 ダウンロード

注意 ダウンロードできるデータの大きさは1つのファイルにつき最大400行までです。また、ダウンロードしたデータのファイルは1つのディスクにつき3個まで作成できます。
ダウンロードするファイルはドライブ名とファイルNo.を指定すると自動的に作成されます。
ダウンロードするファイルが指定したドライブのディスクの空き容量より大きくなった場合、またはダウンロードするファイルが400行より大きくなった場合は、自動的にダウンロードを終了します。

ダウンロード用ファイルは、自動的に次のような名前がつけられます。＊の部分には1～3の数字がつけられます。

ダウンロード用ファイル名：DWFILE　＊

① f・2 キーを押します。

② 表示が次のように変わります。← → キーでダウンロードを行うドライブを選択し、↓ キーを押します。

ダウンロード

ドライブNo.	1	2	3	4	5	6	7	8
ファイルNo.	1			2			3	

↑ ↓ 項目　← → 選択　f・5 実行　ESC 戻り

③ カーソルが「ファイル No.」の項に移動しますので、［←］［→］キーでダウンロードを行うファイルのファイル番号を選択します。
このとき選択した番号のファイルが、すでに指定ドライブ中にあるときは、次のメッセージが表示されます。そのままダウンロードを実行すると、そのファイルの内容はすべて失われ、ダウンロードした内容に書き換えられます。すでに存在するダウンロード用のファイルの内容を失いたくないときは［ESC］キーを押して、ダウンロードするドライブやファイルを選び直します。

指定ファイルは既存のファイルです
上書きしてもよろしいですか

④ ダウンロードするファイルを選択したら、［f・5］キーを押します。
⑤ ダウンロードが実行されます。ダウンロード実行中は、データ通信の内容を画面に表示するとともにファイルに保存します。
ダウンロード中に、ディスクの空き容量が足りなくなった場合は次のメッセージを表示します。これ以上のダウンロードはできませんので［↵］キーを押してダウンロードを終了してください。

ドライブ*のディスクに空き領域がありません
ダウンロードを終了してください

また、ダウンロードしているデータの量が400行を超えた場合は次のメッセージを表示します。この場合もこれ以上のダウンロードはできませんので［↵］キーを押してダウンロードを終了してください。

ダウンロード用ファイルが400行を超えました
ダウンロードを終了します

⑥ ダウンロードを終了する場合は、［f・2］キーを押してください。ダウンロードを終了すると自動的にターミナルモードの画面に戻ります。

10.2.4 アップロード

注意 アップロード用ファイルは通信ユーティリティの「エディタ・モード」で作成します。アップロードできるファイルの大きさは最大400行です。また、アップロード用のファイルは1つのドライブにつき最大3個まで作成できます。

アップロード用ファイルは、自動的に次のような名前がつけられます。*の部分には1～3の数字がつけられます。

**アップロード用ファイル名：UPFILE　*

① [f・3]キーを押します。

② 表示が次のように変わります。[←] [→]キーでアップロードを行うファイルがあるドライブを選択し、[↓]キーを押します。

アップロード								
ドライブ No.	1	2	3	4	5	6	7	8
ファイル No.	1			2			3	

↑ ↓	← →	f・5	ESC
項 目	選 択	実 行	戻 り

③ カーソルが「ファイル No.」の項に移動しますので、[←] [→]キーでアップロードを行うファイルの番号を選択します。
このとき選択したファイルが指定ドライブにないときは、次のメッセージを表示しますので、[←] [→]キーでファイルを選択し直してください。

指定ファイルが見つかりません
もう一度指定してください

④ アップロードするファイルを選択したら、[f・5]キーを押します。

⑤ アップロードが実行されます。アップロード実行中は、画面にアップロードしているデータが表示されています。

⑥ アップロードが終了すると自動的にターミナルモードに戻ります。
アップロードを途中で終了する場合は、[f・3]キーを押してください。アップロードを終了し、自動的にターミナルモードの画面に戻ります。

10.2.5 印刷

① [f・4]キーを押します。

② ①以降、データの送受信を行うたびに通信内容がプリンタに印刷されます。通信内容は今までどうり画面に表示されます。

③ 通信内容の印刷を終了する場合は[f・4]キーを押します。

10.2.6 スクロール・バッファ参照モード

① [HELP]キーを押します。

② 表示が次のように変わります。画面には通信を開始してからの通信内容が表示されています。[↑] [↓]キーまたは[ROLL UP] [ROLL DOWN]キーを使って、画面に表示してる部分を移動して、必要な箇所を表示させます。

参照モード　↑，ROLL DOWN・↓，ROLL UP：上方・下方参照　ESC：通信モードへ

なお、スクロール・バッファに保存されている通信内容は最大で400行までです。通信を開始してからの通信内容が400行を超えた場合は、古い通信内容から自動的に消去します。

③ 通信内容の確認を終了する場合は [ESC] キーを押します。スクロール・バッファ参照モードを終了してターミナルモードの画面に戻ります。

10.3 エディタ・モード

ダウンロードしたファイルや、アップロード用のファイルの編集、作成を行います。

注意 エディタ・モードで編集できるファイルの大きさは最大400行です。なお、この値は1行中の文字数には関係なく、ファイル中に空白の行がある場合もその行を1行として数えますので注意してください。

① メニューから「2　エディタ・モード」を選び、[↵] キーを押します。

② 表示が次のように変わります。[↑] [↓] キーで各項目を選び、[←] [→] キーで編集するファイルの種別、ドライブ番号、ファイル番号を指定します。

編集ファイル	
ファイル種別	アップロード　ダウンロード
ドライブ No.	1　2　3　4　5　6　7　8
ファイル No.	1　2　3
↑ ↓ 項目 / ← → 選択 / f・5 実行 / ESC 戻り	

③ 編集するファイルを指定したら [f・5] キーを押します。

④ 画面が次のように変わります。
画面の入力行に、指定したファイルの1行目のデータが表示されています。
画面の各部の意味は次のとおりです。

表示領域(22行)

入力行(1行)

文頭へ　文末へ　行挿入　行削除　行複写　行貼込　文削除　文読込　印　刷　終　了

• 入力行　：データの編集、作成は入力行に表示されている行に対して行えます。エディタ・モードを起動して、編集するファイルを指定したとき、この行にファイルの一行目が表示されています。

この行には80文字(半角文字)分のデータが入力できます。

• 表示領域：現在入力行に表示されている行よりも前の行の内容がこの領域に表示されます。この領域には22行×80文字分(半角文字)のデータが表示されます。

エディタモードを起動したときは、入力行にファイルの1行目が表示されていますので、この領域にはなにも表示されていません。

10.3.1 編集モードと参照モード

エディタ・モードでは、「編集モード」と「参照モード」の２つのモードがあります。エディタ・モードを起動した直後は、編集モードになっています。

編集モードと参照モードは次のようなときに切り換えて使います。

• 編集モード：入力行に表示されている行にデータを入力するとき。
• 参照モード：入力行に表示されている行を、別の行に変えるとき。

編集モードと参照モードの切り換えの方法は次のとおりです。

• 「編集モード」→「参照モード」：［HELP］キー
• 「参照モード」→「編集モード」：［ESC］キー

• 編集モード

編集モードでは、入力行に表示されている行を編集できます。

ただし、入力行に表示されている行を別の行に変えることはできません。

・参照モード

参照モードでは次のようにして、入力行に表示されている行を移動します。

・ [ROLL UP] キー、 [↓] キー：このキーを押すたびに画面が1行ずつスクロールアップします。

・ [ROLL DOWN] キー、 [↑] キー：このキーを押すたびに画面が1行ずつスクロールダウンします。

参照モードでは、データの入力はできません。

```
                         表示領域(22行)



                          入力行(1行)
[文頭へ][文末へ][行挿入][行削除][行複写]  [行貼込][文削除][文読込][印　刷][終　了]
```

10.3.2 編集、作成の基本操作

ファイルの編集、作成の基本操作は次のとおりです。

・既存ファイルの編集

① エディタ・モードを起動して、編集するファイルのドライブ名と番号を指定します。

② 指定したファイルが画面に表示されます。

この時点では編集モードになっています。画面の入力行にファイルの1行目が表示されています。この行を編集する場合は⑥に進んでください。このとき表示されている行とは別の行を編集する場合は③に進んでください。

③ [HELP] キーを押して参照モードに切り変えます。

④ [ROLL UP] キーまたは [↓] キーを押して、編集したい行を入力行に表示します。

⑤ [ESC] キーを押して、参照モードから編集モードに切り変えます。

⑥ 入力行に表示されている行のデータを次のようにして編集します。

[←] [→] キーで入力行のカーソルを編集したい箇所に移動して、データの削除、挿入をします。

⑦ 入力行のデータを編集したら、 [↵] キーを押します。

編集した行がスクロールアップして表示領域の最下行に移動し、次の行が入力行に表示されます。

これ以降は、③～⑦の操作を繰り返して、ファイルの編集を行います。

・**新規ファイルの作成**

① エディタ・モードを起動して、ファイルのドライブ名と番号を指定します。ファイルを新しく作る場合、作成したファイルはここで指定したドライブと、ファイル番号で保存されます。

② 画面が編集モードになります。新規ファイル作成の場合は、画面にはなにも表示されていません。

③ キーボードから入力行にデータを入力します。

④ データを入力し終わったら、［↵］キーを押します。
画面が1行分スクロールアップして、③で入力した行が表示領域の最下行に移動し、入力行が空白になります。

これ以降は③④の操作を繰り返してデータを入力します。

新規ファイルの作成の場合、入力済のデータは既存ファイルの編集の場合と同様に編集できます。

10.3.3 編集機能

エディタモードではファンクションキーに編集のための機能が設定されています。これらのキーは、編集モード、参照モードのどちらでも使えます。

各キーの機能は次のとおりです。

［f・1］(文頭へ)：編集するファイルの1行目を入力行に表示します。

［f・2］(文末へ)：編集するファイルの最終行を入力行に表示します。また、最終行の直前22行分も表示領域に表示します。

［f・3］(行挿入)：入力行に表示されている行の直前に空き行を1行挿入します。
参照モードでこのキーを押した場合、自動的に編集モードになります。

［f・4］(行削除)：入力行に表示されている行のデータを削除して、メモリ上のペースト・バッファと呼ばれる記憶領域に保存します。ペースト・バッファとは、最大1行分のデータを保存できるデータの一時保管場所です。ペースト・バッファに保存されたデータは［f・6］キーでファイル中に挿入することができます。
［f・6］キーと合わせて使うと行の移動ができます。

［f・5］(行複写)：入力行に表示されているデータをメモリ上のペースト・バッファに保存します。
［f・4］キーと異なり、入力行のデータは削除されません。

［f・6］(行貼込)：［f・4］キー(行削除)、［f・5］キー(行複写)でメモリ上のペースト・バッファに保存された行のデータを、入力行に表示されている行の直前に挿入します。

［f・7］(文削除)：現在編集中のファイルの内容をすべて削除します。

［f・8］(文読込)：入力行に表示されている行のデータの直前にほかのファイルの内容を入力します。

［f・9］(印刷)　：現在編集中のファイルの内容を印刷します。

［f・10］(終了)　：エディタモードを終了します。

これらのファンクションキーを使っての基本的な編集の例を説明します。

以下の説明は、編集するファイルが画面に表示されていて、編集モードの状態であるものとして進めます。

・新しい行を１行追加する場合

① HELP キーを押して参照モードにします。

② ROLL UP ROLL DOWN キーや ↑ ↓ キーで、新しくデータを挿入する行の次の行を入力行に表示します。

③ f・3 キーを押します。

入力行が空白となり、編集モードに切り変わります。

④ 新しいデータを入力します。

⑤ ↵ キーを押します。

新しく入力した行が、②で入力行に表示されていた行の直前に追加されます。

・１つの行を削除する場合

① HELP キーを押して参照モードにします。

② ROLL UP ROLL DOWN キーや ↑ ↓ キーで削除する行を入力行に表示します。

③ f・4 キーを押します。

④ 入力行に表示されていたデータが削除されます。また、入力行には削除した行の次の行が表示されます。ただし、ファイルの最終行を削除した場合は、削除した行の１つ前の行が表示されます。

・行を移動する場合

① HELP キーを押して参照モードにします。

② ROLL UP ROLL DOWN キーや ↑ ↓ キーで移動したい行を入力行に表示します。

③ f・4 キーを押します。

入力行に表示されている行が削除されます。

④ ROLL UP ROLL DOWN キーや ↑ ↓ キーで削除した行を挿入する次の行が入力行に表示されるようにします。

⑤ f・6 キーを押します。

入力行に表示されていた行の直前に③で削除した行が追加されます。

⑥ ESC キーを押して編集モードにします。

・行を複写する場合

① HELP キーを押して参照モードにします。

② ROLL UP ROLL DOWN キーや ↑ ↓ キーで複写したい行を入力行に表示します。

③ f・5 キーを押します。

④ ROLL UP ROLL DOWN キーや ↑ ↓ キーで、複写した行を挿入する次の行を入力行に表示します。

⑤ f・6 キーを押します。

入力行に表示されていた行の直前に③で複写した行が追加されます。

⑥ ESC キーを押して編集モードにします。

・編集中のファイル中にほかのファイルのデータを追加する場合

① HELP キーを押して参照モードにします。

② ROLL UP ROLL DOWN キーや ↑ ↓ キーで、ほかのファイルのデータを追加する直後の行を入力行に表示します。

③　[f・8] キーを押します。

表示が次のように変わりますので、[↑] [↓] キーと [←] [→] キーで追加したいファイルを選択します。

読み込みファイル								
ファイル種別	アップロード				ダウンロード			
ドライブ No.	1	2	3	4	5	6	7	8
ファイル No.	1			2			3	

↑ ↓	← →	f・5	ESC
項　目	選　択	実　行	戻　り

④　[f・8] キーを押します。

入力行に表示されていた行の直前に、新しく指定したファイルの内容が追加されます。

なお、ファイルの追加により、編集中のファイルの大きさが400行を超えた場合は次のメッセージを表示します。[↵] キーを押してファイルの読み込みを終了してください。

編集ファイルが400行を超えるため
文読み込みが実行できません

・編集中のファイルを破棄する場合。

①　[f・7] キーを押します。

②　次のメッセージを表示します。

③　[f・5] キーを押します。

現在編集中のファイルの内容がすべて削除されます。

また、ファイルの編集中に、ファイルの大きさが400行を超えた場合は次のメッセージが表示されます。[↵] キーを押すとエディタ・モードの終了画面になりますので、データの保存、破棄を選んでエディタモードを終了してください。(「10.2.5　編集の終了」を参照してください。)

編集ファイルが400行を超えました
エディタ・モードを終了します

10.3.4　ファイルの印刷

① [f・9] キーを押します。

② 現在編集中のファイルの内容がプリンタに印刷されます。

③ ファイルの印刷中はファイルの編集はできません。

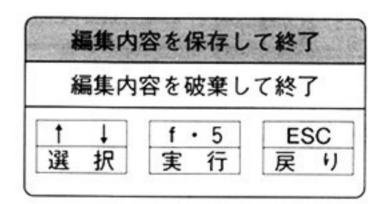
④ ファイルの印刷が終わるとエディタモードに戻り、ファイルの編集ができるようになります。
ファイルの印刷を途中で終了する場合は、[f・9] キーを押してください。

10.3.5　編集の終了

ファイルの編集、作成が終わったらエディタモードを終了します。

・ファイルを保存する場合

① [f・10] キーを押します。

② 次のメッセージが表示されますので、[↑] [↓] キーで「編集内容を保存して終了」を選んで [f・5] キーを押してください。

③ 編集したファイルが、編集前と同じドライブに同じファイル名で保存されます。

ファイル名は次のようになります。＊の部分には1～3の番号がつきます。

・アップロード用ファイル：UPFILE　＊

・ダウンロード用ファイル：DWFILE　＊

「編集内容を保存して終了」を選んだとき、ファイルを保存するドライブの空き領域が足りなくなった場合は表示が次のように変わります。

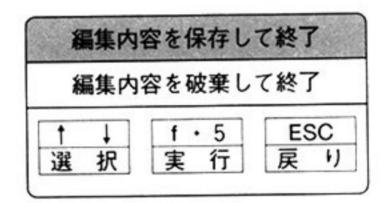
編集、作成したファイルを保存する場合は、表示されているドライブに新しいディスクをセットするか、[←] [↓] キーでほかのドライブを選んで [f・5] キーを押してください。

編集、作成したデータを破棄する場合は ESC キーを押してください。次のメッセージが表示されます。 f・5 キーを押すと編集したデータを破棄してエディタモードを終了します。

編集内容を破棄します
よろしいですか

・ファイルを保存しない場合

① f・10 キーを押します。

② 次のメッセージが表示されますので、 ↑ ↓ キーで「編集内容を破棄して終了」を選んで f・5 キーを押してください。

③ 次のメッセージを表示します。

編集内容を破棄します
よろしいですか

④ f・5 キーを押すと、現在編集中のファイルを破棄してエディタ・モードを終了します。

10.4 通信パラメータ設定

各種の通信パラメータの設定をします。設定されたパラメータはパラメータファイル(ファイル名:PARAF)に保存されます。通信ユーティリティでの通信時には、このパラメータファイルに保存されている設定に従って通信が行われます。

注意 ここで行う通信パラメータの設定はメモリスイッチによる設定とは関係ありません。ここで設定したパラメータの値は、通信ユーティリティで通信を行う場合のみ有効です。
通信パラメータの設定はターミナル・モードでも行うことができますが、この場合設定した値はパラメータファイルには保存されません。ターミナル・モードで設定したパラメータは、ターミナル・モードでの通信中のみ有効です。

ディスク上にパラメータファイル(PARAF)がない場合は、次のページの画面の図のとおりに設定されます。

① メニュー画面から「3　通信パラメータ」を選び、[↵]キーを押します。
② 表示が次のように変わります。
この画面はメニュー部分、ボーレート設定メニューとコマンド入力行の3つの部分に分かれています。

通信ユーティリティ　パラメータ設定

機　能	指　定	
パリティチェック	な　し	奇　数
	偶　数	
データビット長	7ビット	8ビット
ストップビット長	1ビット	1.5ビット
	2ビット	
Xパラメータ	無　効	有　効
Sパラメータ	無　効	有　効
DELコード受信動作	BS	NUL
リターンキー送信処理コード	CR	CR　LF

CR受信処理	CR	CR　LF
日本語シフトコード　KI=	1B4B	1A70
KO=	1B48	1A71
通信方式	全二重	半二重

機　能	指　定		
ボーレート	300	1200	2400
	4800	9600	

コマンド入力	TERM"N81XNBCLP", F

- メニュー部分　　　　　：ボーレートを除く通信パラメータを設定します。
- ボーレート設定メニュー：ボーレートを設定します。
- コマンド入力行　　　　：メニュー部分で設定した項目の値を表示します。また、この行に直接パラメータの値を入力することもできます。

通信パラメータの設定方法は、メニュー画面で各パラメータの値を選ぶ方法と、コマンド入力行で各パラメータの値をキーボードから入力する方法があります。ただし、どちらの場合もボーレートだけはメニューで選ばなければなりません。

なお、メニュー部分のパラメータの値と、コマンド入力行のパラメータの値は連動しています。メニュー部分で指定するとコマンド入力行も同じ設定に変わり、コマンド入力行で指定するとメニュー部分も同じ値に変わります。

10.4.1 メニューからのパラメータの設定方法

① 「通信パラメータ設定」を起動した直後はカーソルが「メニュー部分」の「パリティチェック」の位置にあります。[↑][↓]キーで通信パラメータの項目を選び[←][→]キーで各パラメータの設定を行います。

通信ユーティリティ　パラメータ設定

機　能	指　定	
パリティチェック	な　し	奇　数
	偶　数	
データビット長	7ビット	8ビット
ストップビット長	1ビット	1.5ビット
	2ビット	
Xパラメータ	無　効	有　効
Sパラメータ	無　効	有　効
DELコード受信動作	BS	NUL
リターンキー送信処理コード	CR	CR　LF
コマンド入力	TERM"N81XNBCLP", F	

CR受信処理	CR	CR　LF
日本語シフトコード　KI=	1B4B	1A70
KO=	1B48	1A71
通信方式	全二重	半二重

機　能	指　定		
ボーレート	300	1200	2400
	4800	9600	

② 「メニュー部分」の通信パラメータをすべて設定し終わったら[↓]キーを押します。

③ カーソルが「ボーレート設定メニュー」に移動します。ボーレート設定メニューでは通信速度を設定します。[←][→]キーで通信速度を選択してください。

④ 通信パラメータをすべて設定し終わったら[f・5]キーを押します。
設定した通信パラメータの値はパラメータファイルに保存され、自動的に通信ユーティリティーのメニュー画面に戻ります。

10.4.2 コマンド入力行でのパラメータの設定方法

「コマンド入力行」で通信パラメータを設定する場合、ボーレート以外の項目はキーボードから入力できます。ボーレートだけは「ボーレート設定メニュー」で選択します。

① [↵]キーを押します。

② カーソルが「コマンド入力」の行に移動します。

③ 通信パラメータの設定を入力します。[←][→]キーでコマンド入力行の変更したい箇所にカーソルを移動してパラメータを入力してください。コマンド入力行の記述と、通信パラメータの関係は次のとおりです。

TERM"N81XNBCLP", F(初期設定値)

項目	設定	内容
通信方式	**F**	全二重
	H	半二重
日本語シフトコード	**P**	KI=1B4B, KO=1B48
	I	KI=1A70, KO=1A71
CR 受信処理	C	CR
	L	CR LF
リターンキー送信処理コード	**C**	CR
	L	CR LF
DEL コード受信動作	**B**	BS
	N	NUL
Sパラメータ	**N**	無効
	S	有効
Xパラメータ	N	無効
	X	有効
ストップビット長	**1**	1ビット
	2	1.5ビット
	3	2ビット
データビット長	**7**	7ビット
	8	8ビット
パリティチェック	N	なし
	O	奇数
	E	偶数

④ [↵]キーを押します。カーソルが「ボーレート」の項に移動します。
「ボーレート」の項では通信速度を設定します。[←][→]キーで通信速度を選択してください。

⑤ 通信パラメータをすべて設定し終わったら[f・5]キーを押します。

⑥ 設定した通信パラメータがパラメータファイルに保存され、通信ユーティリティーのメニュー画面に戻ります。

11　最小システムの作成

通常のシステムディスクでBASICを起動する場合は、システムディスクの情報やディップスイッチ、メモリスイッチの設定によりいろいろな機能の拡張を行うことができます。例えば、メモリスイッチ SW6-3が1であれば、機械語モニタのためのシステムが読み込まれ、機械語モニタを使用することができるようになります。

しかし、プログラムによっては機能を拡張する必要はなく、機能が拡張されたためにユーザーエリアが少なくなり、実行できなくなることもあります。このような問題は、BMENUなどを使ってメモリスイッチやシステムディスクの情報を変更して、不要な機能は読み込まないように設定すれば解決します。しかし、別のプログラムを実行する場合には、またメモリスイッチの設定を変更しなければならないなど、面倒な作業が必要になります。このようなときに、メモリスイッチやディップスイッチの設定に関わらず常に必要最低限なシステムだけを読み込むシステムディスクがあれば便利です。この必要最低限なシステムを最小システムと呼びます。このようなシステムディスクと、特定のプログラムを組み合わせておけば、メモリスイッチなどの設定を変更せずにプログラムを実行することができて大変便利です。

最小システムの作成方法

① ユーティリティ SYSSETを実行します。ディスクにはファイル SYSSET.　　　が必要です。

RUN "SYSSET" [↵]

② 次のような画面が表示されます。

```
最小システムの設定処理
システムドライブ NO? 1 [↵]          ②-1
現在の設定：  標準
0：標準システム  ／  1：最小システム
設定するシステムの種類は?1 [↵]       ②-2
標準→最小
設定終了
```

②-1　システムディスクをセットするドライブのドライブ番号を入力します。このドライブにセットしたシステムディスクの設定が変更されます。

②-2　最小システムを設定する場合は1を、もとのシステムに戻す場合は0を入力し [↵] キーを押します。

③　システムディスクに変更内容を書き込み、終了します。

最小システムではメモリスイッチなどの設定に関わらず拡張機能を読み込みません。したがって以下の機能を使用することができません。

カラー表示	8色中8色表示。4096色モードを使用することはできません。
機械語モニタ	機械語モニタの拡張機能は使用できません。
拡張画面ハードコピー	使用できません。標準のハードコピー機能になります。
日本語入力システム	かな漢字変換を用いた日本語入力はできません。コード入力方法で入力することはできます。

付録

A サウンド機能

(PC-286U/US シリーズのみ)

1.　サウンド機能の概要

サウンド機能は、コンピュータを使って音楽を演奏したり、効果音を出す機能です。

PC-286U/US シリーズでは、日本語 Disk BASIC(以下、BASIC と記述します)を使って次のような機能を実行できます。

●FM 音源3和音(チャネル1～3)、SSG 音源3和音(チャネル4～6)の最大6和音の演奏が可能。

●8オクターブの音域をカバー。

●基本音色として82種類の音色を用意。さらに独自の音色を作ることも可能。

●3つのFM 音源にはそれぞれ違った音色を割り当てることが可能。

●ビブラートやトレモロなどの特殊効果の利用が可能。

●他の命令を実行しながらでも演奏が可能。(バックグラウンド演奏)

1.1　サウンド拡張機能について

サウンド機能を使うためには、通常の BASIC 命令のほかにサウンド機能のための拡張機能が必要です。サウンドの拡張機能は、コンピュータ本体に内蔵されている BASIC サポート ROM に格納されています。したがって、コンピュータが正しく設定されていれば、BASIC のバージョンなどに関係なくサウンド拡張機能を使うことができます。

1.2　サウンド機能の仕組みについて

サウンド機能の説明を進めるにあたり、サウンド機能で音の出る仕組み、いくつもの音色を出せる仕組みを簡単に説明しておきます。

●音の高さ、大きさ、音色

音の実体は空気の振動、つまり空気中を伝わる波です。空気中を伝わる波である音の特徴を決めるのは次の3つの要素です。

(1)音程　　音の高さです。音程は波の周波数(振動の速さ)で決ります。

(2)音量　　音の大きさです。音量は波の振幅(振動の大きさ)で決ります。

(3)音色　　音色は色々な周波数の波の集まり方、つまり波の形で決ります。

●FM 音源3チャネル、SSG 音源3チャネル

PC-286U シリーズには FM 音源が3つ、SSG 音源が3つの合計6つの音源が用意されています。これらひとつひとつの音源をチャネルと呼び、チャネル1～3が FM 音源、チャネル4～6が SSG 音源に割り当てられています。6つのチャネルのひとつひとつから、それぞれ違った音程、音量、音色の音を出すことができます。

●SSG 音源

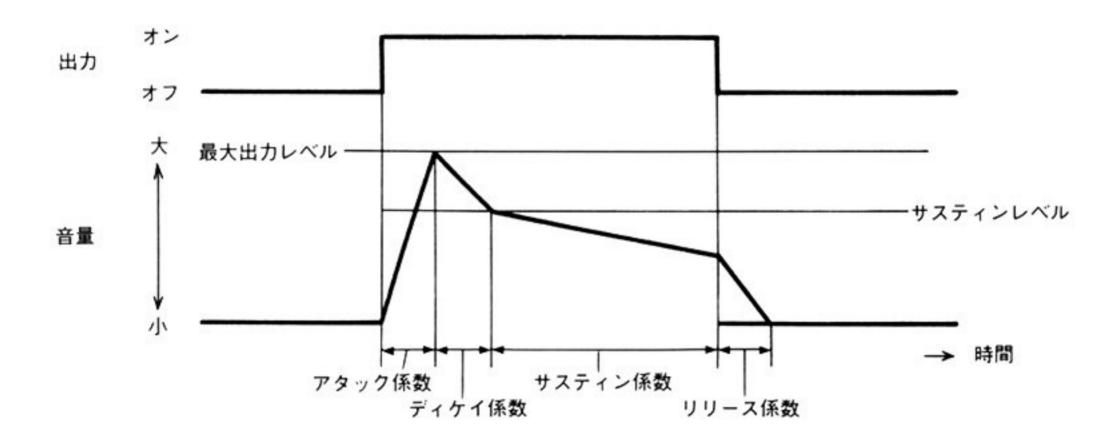
SSG 音源では、音程、音量、音色の中で音色の制御をすることはできません。つまり、SSG 音源に対して波の周波数と振幅を指示すれば出力される音が決ります。

音色の制御ができない SSG 音源の音に表情をつける方法として、エンベロープ・コントロールがあります。エンベロープ・コントロールとは、1つの音に対して振幅(音量)の変化を与えるものです。PC-286U シリーズは、8つの形のエンベロープを持っていて、この形と効果をかける周期の速さによってエンベロープを制御します。

●FM 音源

FM 音源では、音程、音量のほかに音色を制御することができます。PC-286U シリーズには、あらかじめ82種類の音色が音色バンクと呼ばれる場所に登録されており、これらの音色を使ってさまざまな音楽を演奏することができます。さらに独自の音色を作ることもできます。

数多くの音色を作り出すために、FM音源の各チャネルにはそれぞれ4つの演算装置、合計12個の演算装置が用意されています。演算装置は、入力された情報を元にある演算を行い、その演算結果である音(波形)を出力します。ひとつの演算装置から出力された音を次の演算装置に入力し、再び演算を行って新たな音を出力することもできます。音色を決める要素のひとつとして、これらの演算装置をどのように関係付け、それぞれの演算装置にどのような信号を与えるかという設定があります。

演算装置から出力される音の振幅(音量)はおおよそ次のような形をしています。

演算装置に対して、音を出力しなさい(出力のオン)という信号が与えられると、出力する音量を次第に大きくしていきます。この音量の大きくなる割合を決めるのがアタック係数です。

音量がある値になると、今度は音量を次第に小さくしていきます。この限界値になる音量が最大出力レベル、音量が小さくなる割合を決めるのがディケイ係数です。

次に音量がある値まで小さくなると、音量はほぼ一定した値で落ち着きます。音量が一定になる切り替え点はその音量によって定められます。この音量を決めるのがサスティンレベルです。

音量がほぼ一定になるといっても、わずかずつ音量は小さくなっていきます。この音量が小さくなる割合を決めるのがサスティン係数です。

そして、演算装置に出力を終わりなさい(出力のオフ)という信号が与えられると、音量は次第に小さくなってやがて出力が止まります。ここで音量が小さくなる割合を決めるのがリリース係数です。

この音に、さらにピッチ変調、振幅変調、キーボードレイトスケーリング、デチューン係数、マルチプルの効果を加えることができます。

ピッチ変調とは、出力する周波数(音程)を変化させることによって音にビブラートのような効果を与えます。ピッチ変調によるビブラート効果は、その形、速さ、変化の大きさとその効果をかけるタイミングで制御します。

振幅変調は、出力する振幅(音量)を増減することによって音にトレモロのような効果を与えます。振幅変調によるトレモロ効果は、その形、速さ、変化の大きさとその効果をかけるタイミングで制御します。

また、ピアノやギターなど実際の楽器では、同じ楽器でも音程によって音の感じが違っています。これは、音量の増減の割合が違っているからです。そこでサウンド機能では、同じ音色でもその音程によってアタック係数、ディケイ係数、サスティン係数、リリース係数を少しずつ調整しています。この調整の割合を決めるのがキーボードレイトスケーリングです。

さらにいくつもの音を同時に演奏するとき、それらの周波数を微妙に違えてやると、音にうねり効果や深みを与えることができます。この微妙な周波数のずれの設定を決めるのがデチューン係数です。

ある周波数の音にその整数倍の周波数の音を重ね合わせると、さらに違った音色を作ることができます。演算装置では入力された音の周波数に対して整数倍の周波数を持った音を出力することができます。入力された周波数に対して何倍の周波数の音を出力するかを決めるのがマルチプルです。

音色を決めるこれらの条件は音色パラメータと呼ばれ、音色バンクに登録されています。音色パラメータには、それぞれのチャネルに対する設定と、チャネルの中の演算装置ひとつひとつに対する設定とがあります。音色バンクに登録されている音色パラメータは、5×10(第1添字:0～4/第2添字:0～9)の大きさを持った整数型配列を使って読み書きします。

これらの効果を組み合せることによって、限りない音色作りが可能です。

2. サウンド機能を使用するための準備

サウンド機能を使って音楽などを演奏するためには、次のような準備が必要になります。

2.1 機種

ここで説明しているサウンド機能は、PC-286U/USシリーズで使用することができます。

2.2 設定

サウンド機能を使用するためには、メモリスイッチ SW 4 のビット 3 とディップスイッチの SW3-5を次のように設定します。このスイッチを設定することによって、BASIC でサウンド機能を使うことができます。

メモリスイッチ SW 4-3	ディップスイッチ SW 3-5	意　　味
1	OFF	サウンド機能を使用する。
1	ON	BASIC は起動しない。
0	OFF	サウンド機能を使用しない。
0	ON	サウンド機能を使用しない。

□：工場出荷時の設定

注意　メモリスイッチ SW4-3を 1 に設定した場合、ディップスイッチ SW3-5を ON にすると BASIC を起動できません。
PC-286U/US シリーズは、ディップスイッチ SW2-5を OFF にして起動すると、メモリスイッチ SW4-3は 1 に設定されます。

3.　サウンド拡張命令

BASIC では、サウンド機能を使うために次の拡張命令が用意されています。

ON PLAY GOSUB	PLAY 割り込みによる処理ルーチンの定義
PLAY	音楽の演奏
PLAY ALLOC	プレイバッファの確保と初期化
PLAY CLEAR	プレイバッファのクリア
PLAY ON/OFF/STOP	PLAY 割込の制御
STATUS PLAY	未演奏音楽データのバイト数を返す
VOICE	音色バンクの定義
VOICE COPY	音色パラメータの複写
VOICE INIT	音色バンクの初期化
VOICE LFO	ビブラート、トレモロなどの効果をつける
VOICE REG	シンセサイザ LSI のレジスタへの値の設定

これらの命令を使うためには、まず最初に CLEAR 文で機械語プログラム領域を確保し、続いて PLAY ALLOC 文を実行してサウンド拡張命令のための領域を確保する必要があります。PLAY ALLOC 文を実行する前に他の拡張命令を実行しようとしても Illegal function call（違法関数呼び出し）エラーになります。

さらに、これらの命令を使うためには、メモリスイッチとディップスイッチが正しく設定されていることが必要です。メモリスイッチの設定が間違っていて、サウンド機能が使用できない状態にあるときにこれらの命令を実行しようとすると、Syntax error（文法の間違い）のエラーになります。

ON PLAY GOSUB

【オン・プレイ・ゴーサブ】

ON PLAY GO to SUBroutine
PLAY 文の割り込みによってサブルーチンを実行すること。

機　能　PLAY 割り込み処理ルーチンの開始行を定義します。

書　式　ON PLAY(チャネル番号, 残りバイト数)　GOSUB　行番号

使用例　ON PLAY(2,512) GOSUB 9000　←チャネル2のプレイバッファ内の未演奏音楽データが512バイト以下になったときに9000行の割り込み処理ルーチンを実行します。

解　説

- PLAY 文によってバックグランド演奏を実行しているときに、**チャネル番号**で指定したチャネルのプレイバッファ内の未演奏音楽データが**残りバイト数**以下になったときに実行する処理ルーチンを定義します。
- **チャネル番号**には1～6のいずれかのチャネルを指定するか、0または負の数を指定します。
- **チャネル番号**に1～6を指定すると、指定したチャネルの残りバイト数が**残りバイト数**以下になったときに処理の分岐を行います。
- **チャネル番号**に0に指定すると、チャネル1～6の残りバイトの中で最大のものが**残りバイト数**以下になったときに処理の分岐を行います。
- **チャネル番号**に負の数を指定すると、チャネル1～6の残りバイトの中で最小のものが**残りバイト数**以下になったときに処理の分岐を行います。

参　照　PLAY ON/OFF/STOP(PLAY 文による割り込み処理ルーチンの制御)

プログラム例

```
100 ' ON PLAY / PLAY ON / PLAY STOP
110 ' ---  ＰＬＡＹ割り込みで描く図形を切り替える  ---
120 CLEAR ,&H9E00
130 CONSOLE 0,25,0,1:SCREEN 3,0:COLOR ,,,,2:CLS 3
140 PLAY ALLOC 255,255,255,255,255,255
150 GOSUB *MUSIC
160 ON PLAY (-1,14) GOSUB *EN
170 H=RND*639
180 I=RND*399
190 J=RND*639
200 K=RND*399
210 L=(RND*100 MOD 15)+1
220 M=(RND*100 MOD 15)+1
230 LINE(H,I)-(J,K),L,BF,M
240 GOTO 170
250 *MUSIC
260 PLAY STOP
270 CLS 3
280 PLAY CLEAR
290 PLAY "T140"
300 PLAY "MB@1O5E2F#GF#4.E8E2 E2DO4B8O5D8E2R","MB@1O4E2F#GF#4.E8E2 E2DO3B8O4D8E2
R","MB@1O3EEEEEEEE EEEEEEE","MBV15O3BBBBO4CCCC O3BBBBBBB","MBV15O3GGGGAAAA GGGGG
GG","MBV15O3R1F#F#F#F# R1R2."
310 PLAY "MBEE2F#GF#4.E8E2 E(GEG)4BO4BO5E2.R","MBEEE2F#GF#4.E8E2 ERBO3BO4E2.R","
MBEEEEEEEE EED#D#EEER","MBBBBBBO4CCCC O3BBBABBBR","MBGGGGAAAA GGF#F#GGGR","MBR
R1F#F#F#F#R1R1"
320 PLAY ON
330 RETURN 160
340 *EN
350 ON PLAY (0,0) GOSUB *END.P
360 PLAY ON
370 H=RND*639
380 I=RND*399
390 J=RND*100
400 K=(RND*100 MOD 15)+1
410 L=(RND*100 MOD 15)+1
420 CIRCLE (H,I),J,K,,,,F,L
430 GOTO 370
440 *END.P
450 END
```

PLAY

【プレイ】

PLAY
演奏するの意味から、音楽を演奏すること。

機　能　音楽の演奏を行います。

書　式　PLAY　[モード番号,] [文字列1] [, [文字列2] [, [文字列3] [, [文字列4] [, [文字列5] [, 文字列6]]]]]

使用例　PLAY #0, "L8O3CDEFGAB"　←楽音モードで文字列によって指定された音楽を演奏します。

解　説

- FM音源の演奏モードを0～2の**モード番号**で指定します。

#0	楽音モード　：音楽の演奏をするのに適したモードです。
#1	効果音モード：効果音を発生するのに適したモードです。
#2	CSMモード　：サイン波合成法によって音を出します。

- **文字列**1～3はチャネル1～3のFM音源、**文字列**4～6はチャネル4～6のSSG音源に対応します。
- **文字列**1～6は、演奏する音を次に説明するコマンドを使って記述します。これをMML(ミュージック・マクロ・ランゲージ)と呼びます。
- MMLの一部を文字変数や数値変数で記述する場合、次の書式に従って記述します。
  数値　："="+(数値変数名)+"；"
  文字列："X="+(文字変数名)+"；"

CDEFGABコマンド

- 指定した音程の音を出します。コマンドと音程は次のように対応します。
  C=ド　D=レ　E=ミ　F=ファ　G=ソ　A=ラ　B=シ
- 音の半音階は+，#，－、オクターブはOコマンドで指定します。
- 音程の後ろに音長を表す数字をつけることができます(範囲：1～64)。省略時の音長はLコマンドで指定された音長になります。
- 音長は1小節をいくつに分割するかで指定します。
  (例) 1=全音符　4=4分音符
- 符点音符(符点2分音符、符点4分音符など)は音長を表す数字の後に「.」を付けて指定します。

Mx コマンド (初期値：255/範囲：1～65535)

- SSG 音源にエンベロープ効果のかかる周期の速さ(エンベロープ周期)を与えます。(チャネル1～3に対しては無効)
- エンベロープ周期は次の式で与えられます。

　エンベロープ周期(msec)＝(256/667)×設定値

Sx コマンド (初期値：1/範囲：0～15)

- SSG 音源にかかるエンベロープ効果の音量変化の形状(エンベロープ形状)を指定します。(チャネル1～3に対しては無効)

x	エンベロープ形状	x	エンベロープ形状
0，1，2，3，9		11	
4，5，6，7，15		12	
8		13	
10		14	

エンベロープ周期

Vx コマンド (初期値：(チャネル1～3) 0/範囲：0～15)
(チャネル4～6) 8

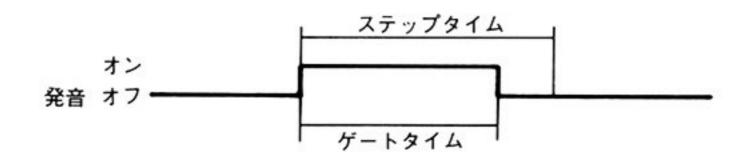
- 音量を設定します。
- 0が最小、15が最大の音量になります。

Lx コマンド (初期値：4/範囲：1～64)

- 音長を指定しない場合の音長を設定します。
- 音長は1小節をいくつに分割するかで指定します。

　(例) 1＝全音符　4＝4分音符
- 符点音符や符点休符(符点2分音符、符点4分休符など)は音長を表す数字の後に「.」を付けて指定します。

Qx コマンド (初期値：7/範囲：1～8)

- ステップタイムに対するゲートタイムの長さの比を設定する。
- ステップタイムとは、音長で設定されている時間です。
- ゲートタイムとは、実際に音を出している時間です。

Qx	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8
ゲートタイム/ステップタイム	1/8	2/8	3/8	4/8	5/8	6/8	7/8	8/8

>，<コマンド

- オクターブの上下を指定します。
- >コマンドで現在のオクターブが1オクターブ上がります。
- <コマンドで現在のオクターブが1オクターブ下がります。
- >,<コマンドを2つ以上続けて指定することによってオクターブを2オクターブ以上、上げたり下げたりすることができます。

Ox コマンド (初期値：4/範囲：1～9)

- オクターブ値を設定します。
- 値が小さいほど音は低く、大きいほど音は高くなります。

Kxコマンド (初期値：なし/範囲：0～96)

• 数値で指定した高さの音を出します。
• FM音源(チャネル1～3)については、K96はK0と同じ高さを音を出します。

音程 ＼ オクターブ	O1	O2	O3	O4	O5	O6	O7	O8	(O9)
C (ド)	0	12	24	36	48	60	72	84	96
C#, D−	1	13	25	37	49	61	73	85	
D (レ)	2	14	26	38	50	62	74	86	
D#, E−	3	15	27	39	51	63	75	87	
E (ミ)	4	16	28	40	52	64	76	88	
F (ファ)	5	17	29	41	53	65	77	89	
F#, G−	6	18	30	42	54	66	78	90	
G (ソ)	7	19	31	43	55	67	79	91	
G#, A−	8	20	32	44	56	68	80	92	
A (ラ)	9	21	33	45	57	69	81	93	
A#, B−	10	22	34	46	58	70	82	94	
B (シ)	11	23	35	47	59	71	83	95	

Txコマンド (初期値：120/範囲：1～255)

• 演奏のテンポを設定します。
• 2つ以上のチャネルに違ったテンポを設定した場合、最後に設定したテンポが有効になります。

Rx, Pxコマンド

• 休符を表します。つまり音の出力を休止します。
• Rx, Pxコマンドの後ろに休符の長さを表す数字をつけることができます(範囲：1～64)。省略時の休符の長さはLコマンドで指定された長さになります。
• 休符の長さは1小節をいくつに分割するかで指定します。
  (例) 1＝全休符　4＝4分休符
• 符点休符(符点2分休符、符点4分休符など)は休符の長さを表す数字の後に「.」を付けて指定します。

＋, −, #

• ＋, #は、音程を示すCDEFGABの後に付けて音程を半音階上げることを表します。
• −は、CDEFGABの後に付けて音程を半音階下げることを表します。

^

・この記号の前後の音をつないで一つの音として出します。(タイ、スラー)

.

・音長や休符の長さを表す数字の後に付けて、符点音符や符点休符を表します。

{・・・} x

・連符を表します。
・xで示す音長の間に { } 内の音を等分に分割して演奏します。

Yr, d コマンド

・サウンド機能を制御する LSI のレジスタに値を直接設定します。
・rに値を設定するレジスタのレジスタ番号、d に設定する値を入力します。
・間違った設定をするとプログラムが正常に動作しなくなる場合があります。レジスタの設定は十分注意して行ってください。
(VOICE REG 文参照)

@Vx コマンド

(初期値：なし/範囲：0～127)

・FM 音源から出る音の音量を設定します。(チャネル 3 ～ 6 に対しては無効)
・V コマンドによる設定と比べて、よりきめ細かい設定ができます。

@Wx コマンド

・コマンド実行の待ち時間を作ります。
・待ち時間の長さ x は、音長と同じ方法で設定します。
・通常は Rx, Px コマンドと同様の働きをします。

@x コマンド

(初期値：0/範囲：0～81)

・FM 音源から出る音の音色の音色番号を設定します。(チャネル 3 ～ 6 に対しては無効)
・チャネル 1 ～ 3 に、それぞれ別の音色を設定することができます。
・音色の初期状態は次表のとおりですが、VOICE コマンドによって独自の音色を作ることもできます。

音色番号	説明
0	音色番号11(ハープシコード)と同じ
1	金管楽器(低音域、音色番号21と同じ)
2	弦楽器(高音域、音色番号25と同じ)
3	エレクトリックピアノ(柔らかい音色、音色番号26と同じ)
4	シセサイザベース(音色番号30と同じ)
5	エレクトリックオルガン(堅い音色、音色番号32と同じ)
6	パイプオルガン(低音域、音色番号34と同じ)
7	フルート(O 5が最適、音色番号36と同じ)

音色番号	説明
8	オーボエ(音色番号38と同じ)
9	クラリネット(音色番号39と同じ)
10	ビブラフォン(音色番号41と同じ)
11	ハープシコード(音色番号46と同じ)
12	ベル(音色番号47と同じ)
13	ピアノ
14	人魂(ひとだま)の飛ぶ音
15	ものが落下する音
16	宇宙船の飛ぶ音
17	ラジコンの走る音
18	効果音(スペースドーム)
19	効果音(インベーダー)
20	音叉(おんさ)
21	金管楽器(高音域)
22	金管楽器(低音域)
23	トランペット(O 5が最適)
24	弦楽器(低音域)
25	弦楽器(高音域)
26	エレクトリックピアノ
27	エレクトリックピアノ(堅い音色)
28	エレクトリックピアノ(柔らかい音色)
29	エレキギター(O 5が最適)
30	シンセサイザベース
31	ウッドベースに近いシンセサイザベース(O 3が最適)
32	エレクトリックオルガン(堅い音色)
33	エレクトリックオルガン(柔らかい音色)
34	パイプオルガン(低音域)
35	パイプオルガン(高音域)
36	フルート(O 5が最適)
37	ピッコロ
38	オーボエ
39	クラリネット
40	グロッケン
41	ビブラフォン
42	シロフォン
43	琴
44	チター
45	クラビネット
46	ハープシコード
47	ベル
48	ハープ(O 5が最適)
49	ピアノ+ベース
50	ハーモニカ
51	ゴムを弾く音
52	ティンパニー(O 3が最適)
53	蒸気機関車
54	アメリカのパトカー

音色番号	説明
55	板ノコギリの音
56	雨だれ
57	ホルン
58	スネアドラム
59	メトロノーム
60	おもちゃのピアノ
61	空缶をたたく音
62	チャイム(O 6が最適)
63	ギターのミュート奏法
64	チェロ
65	チューバ
66	オルガン
67	ギター
68	フルート＋ギター
69	スチールギター
70	エレクトリックベースギター
71	エレクトーン
72	効果音(トンネルの中)
73	効果音(洞窟の中)
74	水滴が弾ける音
75	霧笛
76	ウッドベース
77	大正琴
78	風で鳴る電線の音
79	アコーディオン
80	ピアニカ
81	おばけの足音

＿x コマンド

- x(A～G)で指定した音がC(ド)となるように移調します。
- 指定された音程に対応するキー番号(Kコマンドで指定する音程の番号)に一定の数値を加えて、そのキー番号の音程の音を出します。キー番号に加える数値は指定調によって次のように決められています。

指定調	数値
F♯, G－	－6
G	－5
G♯, A－	－4
A	－3
A♯, B－	－2
B, C－	－1

指定調	数値
C, B♯	0
C♯, D－	1
D	2
D♯, E－	3
E, F－	4
F, E♯	5

＊コマンド

- SSG 音源のエンベロープ周期とエンベロープ形状の指定を解除します。
  (チャネル１～３に対しては無効)

！コマンド

- ＊コマンドで解除した SSG 音源のエンベロープ周期とエンベロープ形状を再設定します。(チャネル１～３に対しては無効)

Zp, v コマンド

- 音色として設定されていない音色を作ります。
- p は VOICE 文や VOICE COPY 文で使用する音色用配列要素の添字と次のように対応します(pの範囲: 0 ～49)。音色用配列要素については VOICE 文を参照してください。
  p＝(第１添字)＋(第２添字)×５
- v はパラメータの内容を示します。(v の範囲: 0 ～16383)
- v に負の数を指定すると、自動的に数値を次のように変換して実行します。
  (設定される値)＝16383＋(指定した負の数)

MF, MB コマンド

- PLAY 文の演奏形態を指定します。
- MF を指定すると、一つのPLAY 文の実行が終わるまで次の命令は実行しません。
- MB を指定すると、PLAY 文の情報をプレイバッファに格納して、すぐに次の命令を実行します。つまり、演奏をしながら他の命令を実行するバックグランド演奏の実行を指定します。

プログラム例

```
100 ' PLAY / PLAY CLEAR
110 ' ---  四角形を描きながらバックグラウンドで音楽を演奏する  ---
120 CLEAR ,&H9E00
130 CONSOLE 0,25,0,1:SCREEN 3,0:COLOR ,,,,2:CLS 3
140 PLAY ALLOC 255,255,255,255,255,255
150 GOSUB *MUSIC
160 ON PLAY (-1,0) GOSUB *END.P
170 H=RND*639
180 I=RND*399
190 J=RND*639
200 K=RND*399
210 L=(RND*100 MOD 15)+1
220 M=(RND*100 MOD 15)+1
230 LINE(H,I)-(J,K),L,BF,M
240 GOTO 170
250 *MUSIC
260 PLAY STOP
270 CLS 3
280 PLAY CLEAR
290 PLAY "T140"
300 PLAY "MB@1O5E2F#GF#4.E8E2 E2DO4B8O5D8E2R","MB@1O4E2F#GF#4.E8E2 E2DO3B8O4D8E2
R","MB@1O3EEEEEEEE EEEEEEE","MBV15O3BBBBO4CCCC O3BBBBBBB","MBV15O3GGGGAAAA GGGGG
GG","MBV15O3R1F#F#F#F# R1R2."
310 PLAY "MBEE2F#GF#4.E8E2 E(GEG)4BO4BO5E2.R","MBEEE2F#GF#4.E8E2 ERBO3BO4E2.R","
MBEEEEEEEE EED#D#EEER","MBBBBBBO4CCCC O3BBBABBBR","MBGGGGAAAA GGF#F#GGGR","MBR
R1F#F#F#F#R1R1"
320 PLAY ON
330 RETURN 160
340 *END.P
350 END
```

PLAY ALLOC

【プレイ・アロケーション】

PLAY ALLOCation
allocation は位置の意味。すなわち演奏のための場所(プレイバッファ)を確保すること。

機能　プレイバッファを確保し、これを初期化します。

書式　PLAY ALLOC　[チャネル1バッファサイズ][,[チャネル2バッファサイズ][,[チャネル3バッファサイズ][,[チャネル4バッファサイズ][,[チャネル5バッファサイズ][,[チャネル6バッファサイズ]]]]]]

使用例　PLAY ALLOC 128, 128, 128　←チャネル1～3のプレイバッファに128バイトずつを確保。

解説

- この命令は、サウンド機能の拡張命令を使用する前に必ず実行してください。
- 1～6の各チャネルの音楽演奏用のプレイバッファを確保します。(単位:バイト)
- チャネル1～3はFM音源、チャネル4～6はSSG音源に対応します。
- **バッファサイズ**に奇数バイトを指定すると、その数に1を加えたバイト数をバッファサイズに設定します。バッファサイズの合計を計算する場合に注意が必要です。
- **バッファサイズ**を省略したチャネルのバッファサイズは0バイトになります。
- プレイバッファのサイズの合計は、CLEAR文で確保した機械語エリアのサイズ以下でなければなりません。
- プレイバッファの確保と同時に、初期値を次のように設定します。

項目	初期値
SSG音源のエンベロープ形状	S1
SSG音源のエンベロープ周期	M255
SSG音源の音量	V8
音長	L4(R4)
音の長さの割合	Q7
オクターブ	O4
テンポ	T120
FM音源の音色番号	0
音色バンク	立ち上げ時の状態

PLAY CLEAR

【プレイ・クリア】

PLAY CLEAR
clear は一掃する、取り除くという意味。これからプレイバッファを初期化すること。

機　能　プレイバッファをクリアします。

書　式　PLAY CLEAR

使用例

```
PLAY CLEAR
```

←プレイバッファ内の音楽データをクリアします。

解　説

- すべてのチャネルの演奏を中止し、プレイバッファ内に残っている音楽データをクリアします。
- PLAY 文による演奏は STOP 文、あるいは STOP キーの押下によって停止することができます。この場合プレイバッファ内には音楽データがそのまま残ります。このままの状態で次の PLAY 文を実行すると、プレイバッファ内に残っている音楽データも演奏します。**PLAY CLEAR** 文ではこのような状態でプレイバッファ内に残っている音楽データをクリアします。

PLAY ON
PLAY OFF
PLAY STOP

【プレイ・オン】
【プレイ・オフ】
【プレイ・ストップ】

PLAY
PLAY 文による割り込みの制御を行うこと。

機能　PLAY 文による割り込み処理ルーチンの実行を許可、禁止、停止します。

書式

```
PLAY |ON  |
     |OFF |
     |STOP|
```

使用例

PLAY ON	←割り込み処理を実行します。
PLAY OFF	←割り込み処理を禁止します。
PLAY STOP	←割り込み処理を停止します。

解説

- 未演奏音楽データが ON PLAY GOSUB 文で設定した値以下になると、実行する割り込み処理ルーチンの制御を行います。
- 割り込み処理ルーチンは ON PLAY GOSUB 文で定義します。
- **PLAY ON** は割り込み処理ルーチンの実行を許可します。この命令を実行後は、指定したチャネルの未演奏音楽データが指定の値以下になると ON PLAY GOSUB 文で定義した割り込み処理ルーチンを実行します。
- **PLAY OFF** は割り込み処理ルーチンの実行を禁止します。この命令を実行後は、指定したチャネルの未演奏音楽データが指定の値以下になっても割り込みは発生しません。プログラム終了時には必ず PLAY OFF を実行してください。
- **PLAY STOP** は割り込み処理ルーチンの実行を停止します。この命令を実行後は、指定したチャネルの未演奏音楽データが指定の値以下になったことだけを記憶し、割り込み処理ルーチンは実行しません。このあと、PLAY ON を実行すると、先ほど未演奏音楽データが指定の値以下になったことによって割り込み処理ルーチンを実行します。

参照　ON PLAY GOSUB(PLAY 文による割り込み処理ルーチンの定義)

STATUS PLAY

【ステータス・プレイ】

STATUS PLAY
status は状態という意味。これから演奏する音楽の状態を調べること。

機能 プレイバッファの未演奏音楽データのバイト数を返します。

書式 STATUS PLAY(チャネル番号)

使用例 IF STATUS PLAY(1)=512 GOTO 1000 ←チャネル1のプレイバッファの未演奏音楽データが512バイトになったら1000行の処理を実行する。

解説
- 指定チャネルのプレイバッファ内の未演奏音楽データのバイト数を返します。
- **チャネル番号**には、1～6のチャネル番号、または0あるいは負の数を指定することができます。
- **チャネル番号**に1～6を指定すると、指定したチャネルのプレイバッファの未演奏音楽データのバイト数を返します。
- **チャネル番号**に0を指定すると、未演奏音楽データが最大のチャネルの未演奏音楽データのバイト数を返します。
- **チャネル番号**に負の数を指定すると、未演奏音楽データが最小のチャネルの未演奏音楽データのバイト数を返します。

プログラム例

```
100 ' STATUS PLAY
110 ' ---  未演奏音楽データのバイト数を表示する ---
120 CLEAR ,&H9E00
130 PLAY ALLOC 128:CLS
140 PLAY "MB@25O1CDEFGAB>CDEFGAB>CDEFGAB>CDEFGAB>CDEFGAB>CDEFGAB>CDEFGAB"
150 ST=STATUS PLAY(1)
160 IF ST>0 THEN LOCATE 0,0:PRINT "STATUS PLAY =";ST:GOTO 150 ELSE END
```

VOICE

【ボイス】

VOICE
声の意味。音色を作ること。

(機能) 音色を作り、音色バンクに登録します。

(書式) VOICE　音色番号，音色用整数型配列名

(使用例) VOICE 1，NEIRO%　　←整数型配列 NEIRO %のパラメータを音色番号1の音色バンクに登録します。

(解説)
- 音色を登録する音色バンクは0～81の82個ありますが、作った音色を登録できる**音色番号**は1～81です。
- **音色用整数型配列名**で指定する配列は2次元配列で、配列の大きさは5×10(第1添字：0～4/第2添字：0～9)です。
- 配列のそれぞれの要素に設定する値は次のとおりです。

要素(0，0)～(0，9)：共通の設定
要素(1，0)～(1，9)：1番目の演算装置に関する設定
要素(2，0)～(2，9)：2番目の演算装置に関する設定
要素(3，0)～(3，9)：3番目の演算装置に関する設定
要素(4，0)～(4，9)：4番目の演算装置に関する設定

要素(0，0)：セルフ・フィードバック/コネクション(範囲：0～63)

7	6	5	4	3	2	1	0
×	×	D5	D4	D3	D2	D1	D0

(2進数の各ビットに対応)

D2～D0：コネクション
D5～D3：セルフフィードバック

- 各チャネルには、入力信号からある決った出力信号を作り出す演算装置がそれぞれ4つあります。その最初の演算装置に変調度を決めるのがセルフ・フィードバック、4つの演算装置の関係を決めるのがコネクションです。
- 1番目の演算装置は、自分自身の出力信号を入力として使っています。D5～D3のデータで、その自分の変調の度合を決めます。(範囲：(0～7)×8)

D5	D4	D3	変調度
0	0	0	OFF
0	0	1	π/16
0	1	0	π/8
0	1	1	π/4

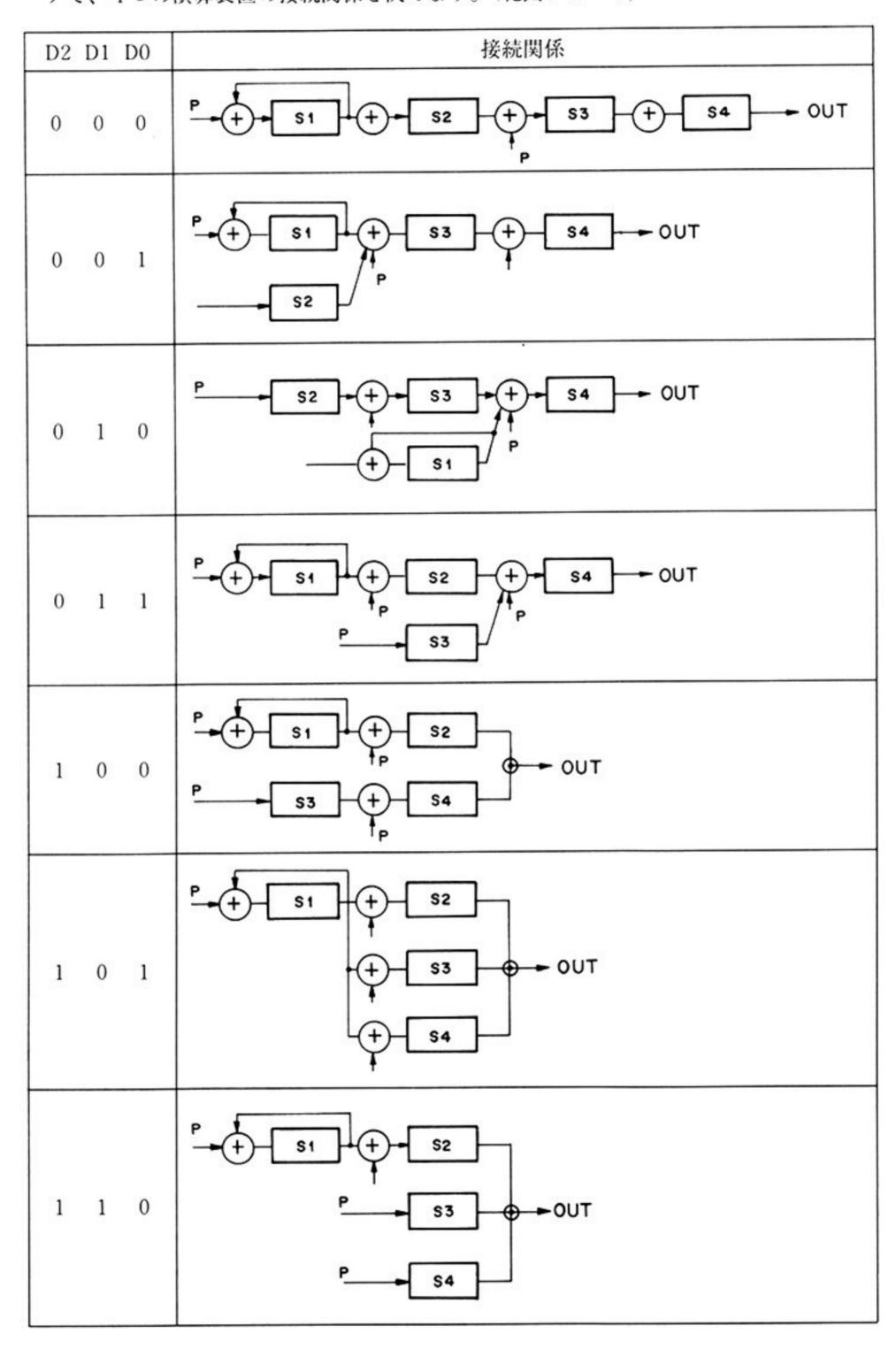
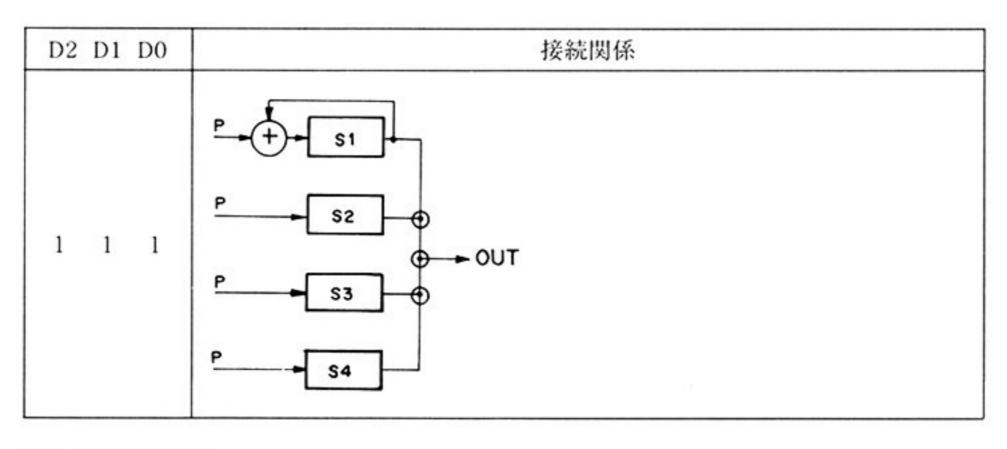
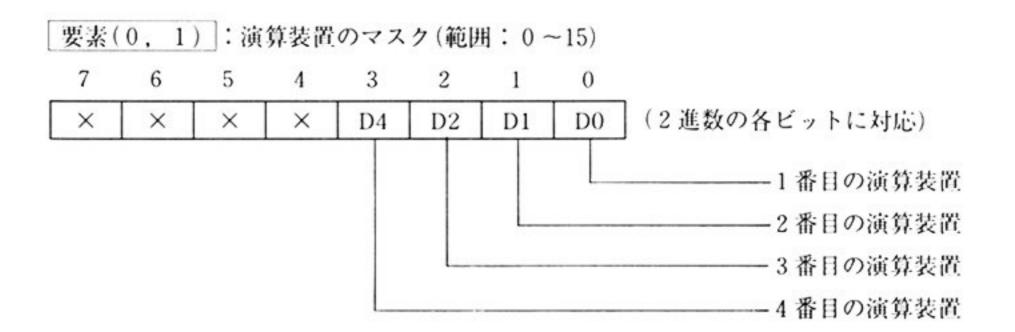
D5	D4	D3	変調度
1	0	0	π/2
1	0	1	π
1	1	0	2π
1	1	1	4π

• 4つの演算装置は、お互いにその出力信号を利用することができます。D2〜D0のデータで、4つの演算装置の接続関係を決めます。(範囲：0〜7)

D2 D1 D0	接続関係
1 1 1	

S1～4：演算装置

P：矩形波の入力

要素(0, 1)：演算装置のマスク(範囲：0～15)

• 各チャネルのそれぞれの演算装置の使用/未使用を決めます。演算装置を未使用にすると、その演算装置に入力された信号がそのまま出力されます。
• 各データが1のとき使用、0のとき未使用に設定されます。

要素(0, 2)：変調波形(範囲：0～5)

• 音に加えるビブラート効果やトレモノ効果の波形を決めます。

0：ノコギリ波

1：短形波

2：三角波

3：サンプル&ホールド

4：ノコギリ波ワンショット

5：三角波ワンショット

(VOICE LFO 文参照)

要素(0, 3)：SYNCディレイ(範囲：0～255)

• 音の鳴り始めとビブラート効果やトレモノ効果をかけるタイミングを設定します。

(VOICE LFO 文参照)

要素(0，4)：ビブラート効果とトレモロ効果の速さ(範囲：0～16383)

• 音に加えるビブラート効果やトレモロ効果の速さを決めます。
• 変形波形で選んだ波形効果のかかる速さを周波数で与えます。

周波数＝(設定値/16383)×250

要素(0，5)：ピッチ変調の深さ(範囲：－127～127)

• 音に加えるビブラート効果の度合を細かく決めます。
• 数値が大きいほどビブラートの効果は大きくなります。
• 負の数を設定すると、波形が反転します。

要素(0，6)：振幅変調の深さ(範囲：－127～127)

• 音に加えるトレモロ効果の度合を細かく決めます。
• 数値が大きいほどトレモロ効果は大きくなります。
• 負の数を設定すると、波形が反転します。

要素(0，7)：ピッチ変調の深さ(範囲：0～15)

• 音に加えるビブラート効果の度合を大まかに決めます。
• 数値が大きいほどビブラート効果は大きくなります。

要素(0，8)：未使用

要素(0，9)：未使用

要素(1，0)/(2，0)/(3，0)/(4，0)：演算装置のアタック係数(範囲：0～31)

• 演算装置のアタック係数を決めます。
• 数値が大きいほど音の立上り時間は短くなります。

要素(1，1)/(2，1)/(3，1)/(4，1)：演算装置のディケイ係数(範囲：0～31)

• 演算装置のディケイ係数を決めます。
• 数値が大きいほど音量の減衰の割合は大きくなります。

要素(1，2)/(2，2)/(3，2)/(4，2)：演算装置のサスティン係数(範囲：0～31)

• 演算装置のサスティン係数を決めます。
• 数値が大きいほど音量の減り方が穏やかになります。

要素(1，3)/(2，3)/(3，3)/(4，3)：演算装置のリリース係数(範囲：0～15)

• 演算装置のリリース係数を決めます。
• 数値が大きいほど出力が止まるまでの時間は短くなります。

要素(1，4)/(2，4)/(3，4)/(4，4)：演算装置のサスティンレベル(範囲：0～15)

• 演算装置のサスティンレベルを決めます。
• 数値が大きいほど切り替えのタイミングが遅くなります。

要素(1，5)/(2，5)/(3，5)/(4，5)：演算装置の最大出力レベル(範囲：0～127)

• 演算装置の最大出力レベルを決めます。
• 数値が大きいほど音量は高いレベルまで増加します。

要素(1，6)/(2，6)/(3，6)/(4，6)：演算装置のキーボードレイトスケーリングの深さ(範囲：0～3)

• 演算装置のキーボードスケーリングを深さを決めます。
• 数値が大きいほど調整の割合が大きくなります。

要素(1，7)/(2，7)/(3，7)/(4，7)：演算装置のマルチプル(範囲：0～15)

• 演算装置のマルチプルを決めます。

値	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
倍率	½	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

要素(1，8)/(2，8)/(3，8)/(4，8)：演算装置のデチェーン係数(範囲：0～3)

• 演算装置のデチェーン係数を決めます。
• 数値が大きいほどずれの割合が大きくなります。
• 負の数を設定すると、周波数のずれの方向が逆になります。

要素(1，9)/(2，9)/(3，9)/(4，9)：演算装置の振幅変調の深さ(範囲：0～15)

• 演算装置に加えるビブラート効果の度合を大まかに決めます。
• 数値が大きいほど振幅変調の度合は大きくなります。

参照　VOICE COPY(音色のバンクのパラメータのコピー)

VOICE COPY 【ボイス・コピー】

VOICE COPY
copy は複写すること。音色のパラメータを複写すること。

機　能　音色バンクの音色パラメータを整数型配列にコピーします。

書　式　VOICE COPY　音色番号，音色用整数型配列名

使用例　VOICE COPY 15，NEIRO%　←音色番号15の音色パラメータを整数型配列NEIRO%にコピーします。

解　説
- **音色番号**で指定した音色バンクに登録されている音色のパラメータを、整数型の配列にコピーします。
- **音色用整数型配列**は２次元配列で、配列の大きさは５×10(第１添字：０～４/第２添字：０～９)です。
- 配列のそれぞれの要素の内容についてはVOICE文を参照してください。

参　照　VOICE(音色の作成と登録)

プログラム例

```
100 ' VOICE COPY / VOICE
110 ' ---  音色バンク6の音色にビブラートをかける  ---
120 CLEAR ,&H9E00
130 PLAY ALLOC 128
140 DIM PARA%(4,9)
150 PRINT "元の音色（音色番号＝４）"
160 PLAY "L2@4O4CDEFGAB>C"
170 VOICE COPY 4,PARA%
180 ' 音色パラメータの設定
190 PARA%(0,2)=2:'三角波
200 PARA%(0,3)=1:'発音と同時にビブラートをかける
210 PARA%(0,4)=200:'周波数=(200/16383)*250
220 PARA%(0,5)=80:'ピッチ変調の深さ（微調整）=80
230 PARA%(0,7)=10:'ピッチ変調の深さ（粗調整）=10
240 VOICE 4,PARA%
250 PRINT:PRINT "ビブラートをかけた音色"
260 PLAY "L2@4O4CDEFGAB>C"
```

VOICE INIT

【ボイス・イニシャライズ】

VOICE INITialize
initialize は初期化するという意味。これから音色バンクを初期化すること。

機　能　音色バンクを初期化します。

書　式　VOICE INIT　[音色番号]

使用例

VOICE INIT　←すべての音色バンクの状態を初期状態にします。

VOICE INIT 10　←音色番号10の音色パラメータを初期状態にします。

解　説

- FM 音源の音色バンクを初期状態に戻します。(VOICE 文の@x コマンドを参照)
- **音色番号**(0 ~81)を指定した場合はその音色番号の音色を、指定しない場合はすべての音色を初期状態に戻します。
- SSG 音源に関する設定は初期化されません。
- このコマンドを実行すると、チャネル1～3の音色には音色番号0の音色が設定されます。
- このコマンドを実行すると、VOICE 文で使用した音色用の整数型配列は解放されます。

VOICE LFO

【ボイス・エル・エフ・オー】

VOICE LFO
LFOとは音に加える効果のこと。音にビブラートやトレモロなどの効果を加えます。

機能　ビブラートやトレモロなどの効果を加えます。

書式　VOICE LFO　チャネル番号 [, [波形番号] [, [SYNC ディレイタイム] [, [速さ] [, [ピッチ変調深さ(微調整)] [, [振幅変調深さ(微調整)] [, ピッチ変調深さ(粗調整)]]]]]]

使用例　VOICE LFO 1, 4, 20, 1000, −5, 1, 6　←チャネル1の音にビブラート効果とトレモロ効果を加えます。

解説

- **チャネル番号**で指定したチャネルから出る音にビブラートやトレモロなどの効果を加えます。
- **波形番号**でビブラート効果やトレモロ効果の種類を選びます。

(縦軸は、ビブラート効果の場合周波数(音程)のずれ、トレモロ効果の場合振幅(音量)の変化を表します。)

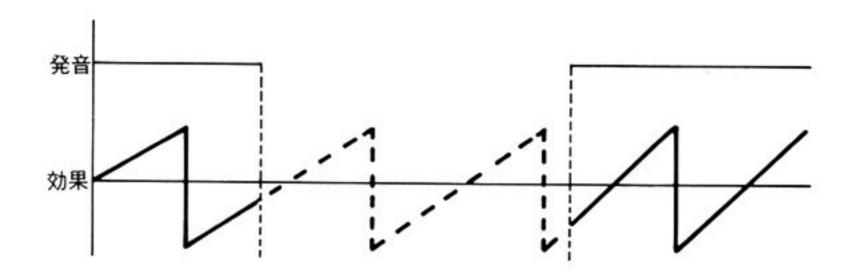
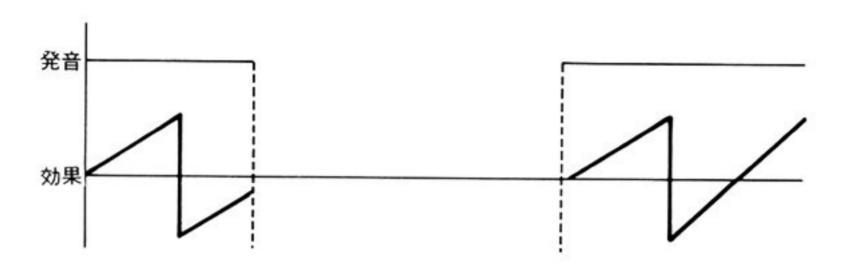
波形番号	波形名称	波形
0	ノコギリ波	
1	矩形波	
2	三角波	
3	サンプル&ホールド	
4	ノコギリ波ワンショット	
5	三角波ワンショット	

一定周期でランダムに変化。(波形番号3)

• **SYNC ディレイタイム**は、音の鳴り始めと効果をかけるタイミングを設定します。

<SYNC ディレイタイムに0を設定したとき>

<SYNC ディレイタイムに1を設定したとき>

<SYNC ディレイタイムに1より大きい数値を設定したとき>

• **速さ**は、効果のかかる周期の速さを0～16383の間の数値で設定します。波形の周波数は次の式で求められます。

周波数＝(速さ/16384)×1000/32

• **ピッチ変調深さ(微調整)**は、ビブラートのかかる深さを設定します。(範囲：－127～127)
• 数値が大きいほどビブラートのかかる深さは大きくなり、負の数を設定すると波形が反転します。
• **振幅変調の深さ(微調整)**は、トレモロ効果のかかる深さを決めます。(範囲：－127～127)
• 数値が大きいほどトレモロ効果のかかる深さは大きくなり、負の数を設定すると波形が反転します。
• **ピッチ変調の深さ(粗調整)**は、ビブラートのかかる深さを大まかに決めます。(範囲：0～15)
• 数値が大きいほどビブラートのかかる深さは大きくなります。

プログラム例

```
100 ' VOICE LFO
110 ' ---  ＬＦＯ効果の比較  ---
120 CLEAR ,&H9E00
130 PLAY ALLOC 128
140 PRINT "ＬＦＯ効果なし"
150 PLAY "T80@20C1"
160 FOR I=0 TO 5
170   PLAY "@20"
180   PRINT "波形番号 =";I
190   VOICE LFO 1,I,1,800,127,127,15
200   PLAY "T80C1"
210   FOR J=0 TO 200:NEXT J
220 NEXT
```

VOICE REG

【ボイス・レジスタ】

VOICE REGister
LSIの設定を記録しておく場所(レジスタ)に値を設定すること。

機　能　サウンド機能を制御するLSIのレジスタに値を直接設定します。

書　式　VOICE REG　レジスタ番号，出力データ

使用例　VOICE REG &H0，1　←レジスタ番号0のレジスタに1を設定します。

解　説

- コンピュータに内蔵されている、サウンド機能を制御するLSIの**レジスタ番号**で指定したレジスタに**出力データ**を直接設定します。
- レジスタの内容は次のとおりです。

レジスタ番号	内　容
0～5	矩形波音源の周波数を与えます。
6	ノイズ源の周波数を与えます。
7	I/Oポートの入出力制御および楽音とノイズの出力を制御します。
8～10	音量を制御します。音量制御には固定音量(プログラマブル)と可変音量とがあります。
11～12	可変音量の場合のエンベロープの周期を制御します。
13	可変音量の場合のエンベロープの形状を指定します。
14～15	8ビットの汎用I/Oポート。
33	テスト情報。常に"0"の状態にしておきます。
36～38	タイマーAとタイマーBのセット時間を与えます。
39	タイマーA・Bの動作を制御します。加えて、FM音源第3チャネルのモードを設定します。
45～47	入力クロックの分周数を指定します。FM音源は2～6分周、矩形波音源は1～4分周です。
48～62	デチューン係数とマルチプルを設定します。
64～78	最大出力レベルを与えます。この情報は、被変調波の音量・変調波の変調指数になります。

レジスタ番号	内　　容
80～94	キーボードレイトスケーリングとアタック係数を設定します。
96～110	ディケイ係数を設定します。
108～126	サスティン係数を設定します。
128～142	サスティンレベルとリリース係数を設定します。
144～158	エンベロープ波形の形状を設定します。
160～166	各チャネルの音程を設定します。
168～174	特殊モードにした場合の3チャネルの音程です。
176～178	1番目の演算装置の変調度と4つの演算装置の接続関係を設定します。

• 直接LSIの制御に関係しているレジスタに値を設定する際には、十分注意してください。特に次のレジスタには値の設定を行わないでください。

33～39

45～47

4. エラーメッセージ

サウンド拡張命令では、次のエラーメッセージを表示します。

エラーメッセージ	エラーメッセージの内容(発生の原因と対策)
Illegal function call 違法関数呼び出し 5	(PLAY ALLOC 文) 確保しようとしたプレイバッファのサイズの合計が65535(&HFFFF)を越えています。
Out of octave オクターブ範囲を越えた 134	(PLAY 文) +, −, #で音程を半音上下したときに演奏できる音程の範囲を越えています。
Sound buffer not allocated サウンドバッファがない 133	(VOICE, PLAY 文) PLAY ALLOC 文を実行せずにサウンド拡張機能を使おうとしています。
Sound buffer overfow プレイバッファの不足 132	(PLAY 文) 音楽データの容量が確保したプレイバッファの容量を越えています。

5. 音色作りのヒント

音色を作る、つまり音色パラメータを決める場合、それぞれの音色パラメータの特徴を知っておく必要があります。ここでは、パラメータの設定とその結果として出力される音の関係がわかりにくいものについてその特徴を説明します。

また、初期状態で音色バンクに設定されている音色パラメータの設定を紹介しますので、音色作りの参考にしてください。

5.1 音色パラメータの特徴

●セルフ・フィードバック

・音の鋭さを変化させます。

・設定値が大きいほど鋭い音になります。

●コネクション

【000, 001, 010, 011】

・一つの音色を中心にして音に変化を与えます。

・比較的明るい音になります。

・単独の楽器の音色を作るのに適した設定です。

【100】

・２つの音色を組み合せた音作りが可能です。

・音のピッチをずらしてコーラス効果を加えたり、まったく違った音色を重ね合わせることもできます。

【101, 110, 111】

・厚みのある落ち着いた音作りに適した設定です。

・違った音色や、少しずつピッチを変えた音を重ね合わせることができます。

●アタック係数/ディケイ係数/サスティン係数/リリース係数/サスティンレベル

・音の丸みを変化させます。

・設定値が小さいほど丸みを帯びた音、大きいほど明るい音になります。

5.2 音色パラメータの例

●ハープシコード(音色番号０)

第２添字＼第１添字	0	1	2	3	4
0	7×8＋2	0	0	0	0
1	15	21	27	21	25
2	0	27	25	26	23
3	0	9	9	9	9
4	0	15	3	15	13
5	0	77	97	97	127
6	0	2	3	6	2
7	0	12	15	1	3
8	0	3	1	－4	－3
9	0	0	0	0	0

・厚みのある音を作りやすいコネクションを採用しています。

・３倍音、12倍音、15倍音を組み合せています。

●シンセサイザーベース(音色番号4)

第2添字＼第1添字	0	1	2	3	4
0	4×8＋0	0	0	0	0
1	15	23	24	23	24
2	0	25	25	25	25
3	0	7	7	7	7
4	0	12	13	14	0
5	0	97	82	107	127
6	0	2	3	2	2
7	0	6	5	0	1
8	0	－3	3	－3	3
9	0	0	0	0	0

・2つの音色を組み合せるコネクションを採用しています。

・セルフ・フィードバックを無効にしています。

・次の演算装置に出力する信号と、最終的に出力する信号で対照的なデチューン係数を設定しています。

●パイプオルガン(低音域、音色番号4)

第2添字＼第1添字	0	1	2	3	4
0	0×8＋4	11	16	13	16
1	15	29	29	29	29
2	0	31	31	31	31
3	0	8	9	10	10
4	0	15	15	15	15
5	0	97	127	97	127
6	0	2	2	2	2
7	0	8	4	0	0
8	0	－1	－1	－2	1
9	0	0	0	0	0

・明るい音の作りやすいコネクションを採用しています。

・ビブラート効果、トレモロ効果を使っています。

・1/2倍音を使っています。

B メモリスイッチ

PC-286 シリーズは電源スイッチがオフの場合でも、バックアップ電池によってデータをバックアップしているメモリを持っています。これにシステムのさまざまな情報を書き込んで利用しているのがメモリスイッチです。

メモリスイッチの変更はいつ行っても構いませんが、メモリスイッチの内容は電源スイッチをオンした直後に読み込まれます。したがって、メモリスイッチを変更した後は、リセットを行うか、一度電源を切り、再び電源を入れるようにしてください。また、メモリスイッチを変更する場合はディップスイッチ SW2-5 を ON にします。SW2-5 が OFF の場合は起動時にシステム設定値を設定します。

メモリスイッチの変更は次の方法で行うことができます。

(1)日本語Disk BASICのユーティリティディスクの中のユーティリティソフト BMENU を使用する。
(2)日本語 Disk BASIC の機械語モニタを使用する。

メモリスイッチの各々の役割はつぎのとおりです。

●表の見方

メモリアドレス	このメモリスイッチを設定しているメモリ上のアドレス
システム設定値	PC-286 シリーズの初期値。ディップスイッチ SW2-5 が OFF のとき、電源立ち上げ時に設定する値です。
機能	メモリスイッチの名称
意味	そのビットが0、1のときの意味を説明します。
&H××××、&H××	16進数を示します。

(1) メモリスイッチ SW 1

メモリアドレス　&HA3FE2

システム設定値　&H48(01001000)

機　能			意　　味
XON／XOFF 制御	0		**XON／XOFF 制御を行わない**
	1		XON／XOFF 制御を行う
未使用	0		**常に0にする**
データビット長	1	0	**7ビット**
	1	1	8ビット
パリティチェック	0		**なし**
	1		あり
パリティ指定	0		**奇数パリティ**
	1		偶数パリティ
ストップビット長	0	1	**1ビット**
	1	0	1.5ビット
	1	1	2ビット

□：システム設定値

【説明】

OPEN 文で通信パラメータを省略した場合は、このメモリスイッチで指定した値が用いられます。

(2) メモリスイッチ SW 2

メモリアドレス　　&HA3FE6

システム設定値　　&H05(00000101)

7 6 5 4 3 2 1 0

機能	意味				
通信速度 (ビット3〜0)	0	0	0	1	75 bps
	0	0	1	0	150 bps
	0	0	1	1	300 bps
	0	1	0	0	600 bps
	0	**1**	**0**	**1**	**1200bps**
	0	1	1	0	2400bps
	0	1	1	1	4800bps
	1	0	0	0	9600bps
日本語シフトコード (ビット4)	**0**	**KI＝1B4Bh、KO＝1B48h**			
	1	KI＝1A70h、KO＝1A71h			
受信時の復帰、改行動作 (ビット5)	**0**	**CR のみ**			
	1	CR＋LF			
↵ キー押下時の送出コード (ビット6)	**0**	**CR のみ**			
	1	CR＋LF			
SI／SO 制御 (ビット7)	**0**	**SI／SO 制御を行わない**			
	1	SI／SO 制御を行う			

□：システム設定値

【説明】

通信速度は OPEN 文では設定することはできません。

(3) メモリスイッチ SW 3

メモリアドレス　&HA3FEA
システム設定値　&H04 (00000100)

7 6 5 4 3 2 1 0

ビット	機　能	意　味			
2-0	使用可能なメモリ容量	0	0	0	128K バイト
		0	0	1	256K バイト
		0	1	0	384K バイト
		0	1	1	512K バイト
		1	**0**	**0**	**640K バイト**
3	数値演算プロセッサの使用（アドバンストモードで起動）	**0**	**数値演算プロセッサを使用しない**		
		1	数値演算プロセッサ使用する		
4	数値演算プロセッサの使用（ノーマルモードで起動）	**0**	**数値演算プロセッサを使用しない**		
		1	数値演算プロセッサ使用する		
5	未使用	**0**	**常に0にする**		
6	立ち上げ時のテキスト画面の文字の色	**0**	**白**		
		1	緑		
7	入力モードでDEL コード受信時動作	**0**	**DEL (7Fh、FEh) コードとする**		
		1	NUL (00h) コードとする		

□：システム設定値

【説明】

- 使用可能なメモリ容量は BASIC で使用できるメモリの大きさを設定するものです。通常は640KB にしておきます。
- SW3-3/4は EPSON PC シリーズで数値演算プロセッサを使用するときに1にします。

【注意】

- PC-286U で数値演算プロセッサを装着したときは CPU クロックを 8 MHz にしてください。
- 数値演算プロセッサを装着していないときに「数値演算プロセッサを使用する」に設定しないでください。

(4) メモリスイッチSW4

メモリアドレス　&HA3FEE

システム設定値　&H00(00000000)　PC-286Uシリーズは&H08(00001000)

7 6 5 4 3 2 1 0

ビット	機能	意味	
0	未使用	0	常に0にする
1	未使用	0	常に0にする
2	拡張ROM装着 &HC8000～&HC9FFF	0	なし
		1	あり
3	拡張ROM装着 &HCC000～&HCFFFF	0	なし(サウンド機能使用しない)
		1	あり(サウンド機能使用する)
4	拡張ROM装着 &HD0000～&HD3FFF	0	なし
		1	あり
5	拡張ROM装着 &HD4000～&HD5FFF	0	なし
		1	あり
6	拡張ROM装着 &HCA000～&HCBFFF	0	なし
		1	あり
7	拡張ROM装着 &HCE000～&HCFFFF	0	なし
		1	あり

□：システム設定値

【解説】

このメモリスイッチは拡張ボードを装着したときに拡張ボード添付の説明書に従って変更してください。

PC-286U/USシリーズは本体標準でサウンド機能を持っていますのでSW4-3のシステム設定値は1になります。サウンド機能を使用しない場合は、Dip SW2-5をONにしてから、SW4-3を0にして、BASICを起動します。

(5) メモリスイッチSW5

メモリアドレス　&HA3FF2

システム設定値　&H01(00000001)

7 6 5 4 3 2 1 0

機能	意味				
使用するプリンタ	0				PC-PR201シリーズ以外のプリンタ
	1				**PC-PR201シリーズのプリンタ**
ドライブ番号の順位指定	**0**				**フロッピーディスクドライブ→ハードディスクドライブの順にドライブ番号を割り当てる**
	1				ハードディスクドライブ→フロッピーディスクドライブの順にドライブ番号を割り当てる
ハードディスクユーザー識別名使用	**0**				**使用する**
	1				使用しない
画面ハードコピー	**0**				**白黒画面ハードコピー**
	1				カラー画面ハードコピー
システムの立ち上げ装置の指定	**0**	**0**	**0**	**0**	**フロッピーディスクドライブ→ハードディスクドライブの順にサーチ**
	0	0	1	0	640KBフロッピーディスクドライブを立ち上げ装置とする(他の装置は読みにいかない)
	0	1	0	0	1MBフロッピーディスクドライブを立ち上げ装置とする(他の装置は読みにいかない)
	1	0	1	0	ハードディスクドライブ1を立ち上げ装置とする(他の装置は読みにいかない)
	1	0	1	1	ハードディスクドライブ2を立ち上げ装置とする(他の装置は読みにいかない)

□:システム設定値

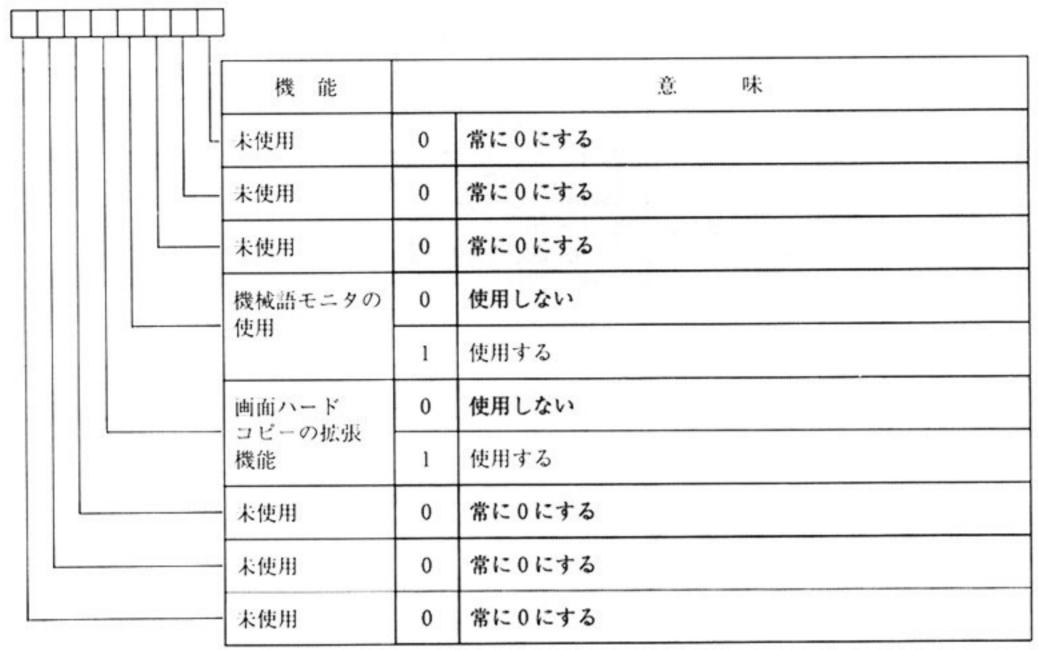
【解説】

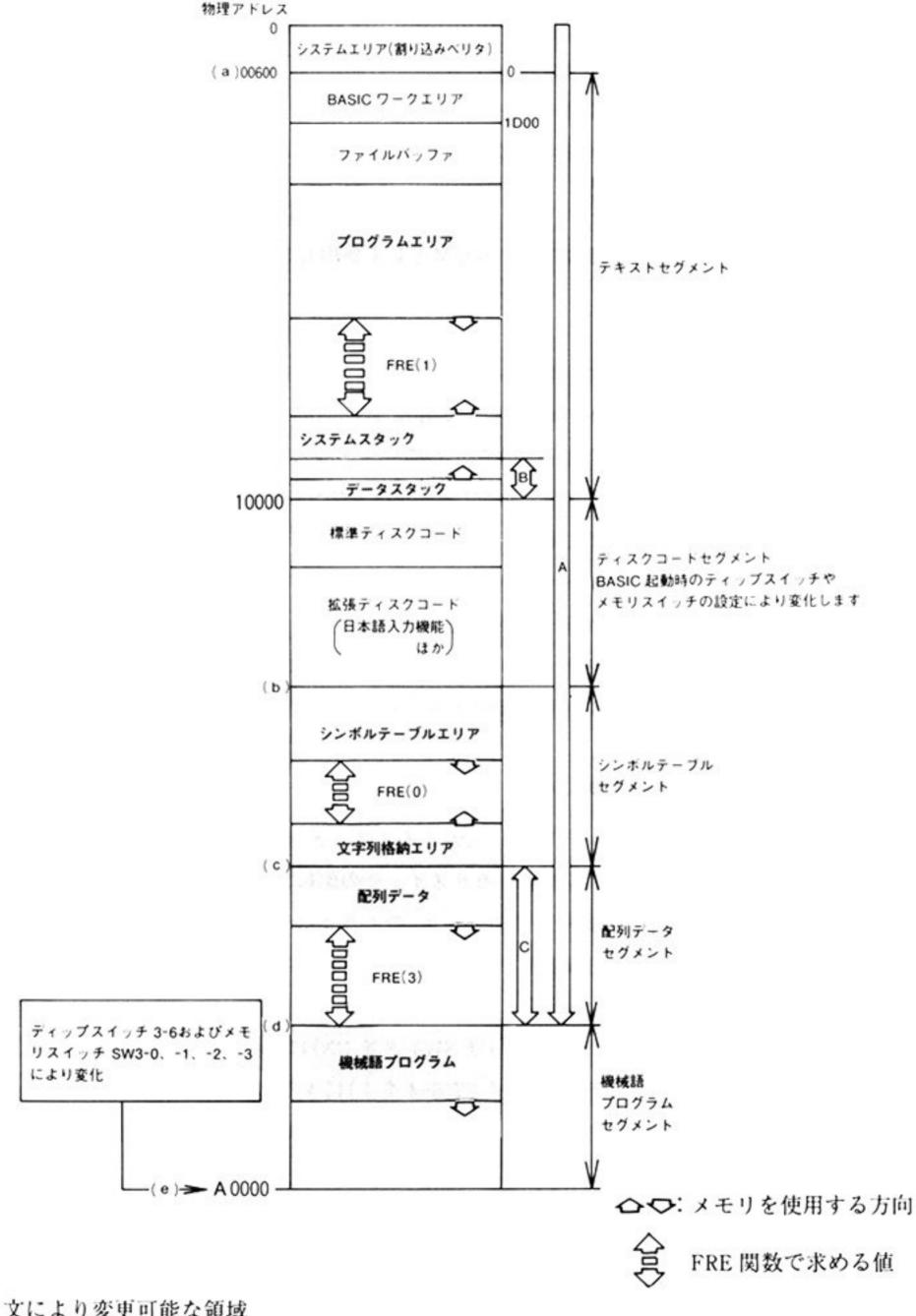
画面ハードコピーはメモリスイッチSW6-4が1で、PC-PR系カラープリンタが接続されている場合にカラーの画面ハードコピーをとることができます。

(6) メモリスイッチSW6

メモリアドレス　　&HA3FF6

システム設定値　　&H00(00000000)

ビット	機　能	意　　味	
0	未使用	0	**常に0にする**
1	未使用	0	**常に0にする**
2	未使用	0	**常に0にする**
3	機械語モニタの使用	0	**使用しない**
		1	使用する
4	画面ハードコピーの拡張機能	0	**使用しない**
		1	使用する
5	未使用	0	**常に0にする**
6	未使用	0	**常に0にする**
7	未使用	0	**常に0にする**

□：システム設定値

C メモリマップ

• CLEAR 文により変更可能な領域

CLEAR [[ダミー], [プログラム領域], [スタック領域], [配列変数領域]]

A　B　C

Aの初期設定　ユーザーメモリの最大値(&HA0000)

Bの初期設定　512バイト

Cの初期設定　ディスクコードの大きさにより変化

• 各セグメントアドレスは、次のアドレス(セグメント &H0060 からのオフセット)に格納されています。

&H06A2：配列データセグメント……………………(c)

&H1402：ユーザーメモリの上限……………………(e)

&H1404：機械語プログラムセグメント…………(d)

&H140E：テキストセグメント………………………(a)

&H1410：シンボルテーブルセグメント…………(b)

システムエリア

割り込みベクタおよび入出力制御のワークエリアとして使用します。

(1) テキストセグメント

BASIC ワークエリア　BASIC インタープリタのワークエリアです。

ファイルバッファ　BASIC 起動時の"How many files?"に答えた数のファイルバッファを用意します。一つのファイルバッファはファイル制御ブロック(40バイト)およびファイルバッファ(256バイト)から構成されます。

プログラムエリア　BASIC プログラムを格納するエリアです。

システムスタック　BASIC インタープリタが使用するスタックエリアです。

データスタック　FOR～NEXT 文や GOSUB 文など BASIC プログラムを実行する際に使用するスタックエリアです。CLEAR 文で変更することができます。BASIC 起動時の設定は512バイトです。

(2) ディスクコードセグメント

日本語 Disk BASIC のシステムプログラム(ディスクコード)をロードするエリアです。システムプログラムの大きさはディップスイッチやメモリスイッチの値により異なります。

(1)標準のシステムプログラム(標準ディスクコード)の大きさは 24KB です。

(2)日本語入力機能の大きさは約 150KB です。日本語入力機能はユーティリティソフト BMENU により切り離すことができます。

(3)拡張グラフィックモード(ディップスイッチ SW1-8 を ON)にすると約 18KB 大きくなります。

(4)画面ハードコピーの拡張(メモリスイッチ SW6-4 を 1)にすると約 6KB 大きくなります。

(5)機械語モニタを使用(メモリスイッチ SW6-3 を 1)にすると約 25KB 大きくなります。

(3) シンボルテーブルセグメント

シンボルテーブルエリア　シンボルテーブルセグメント全体のサイズは配列変数領域の増減により変化します。ただし、64KBを超すことはありません。
シンボルテーブルセグメントにはラベル、変数名、関数名とその属性、数値変数のデータおよび文字変数のストリングディスクリプタを格納します。

文字格納エリア　ストリングディスクリプタで示す実際の文字データを格納します。

(4) 配列データセグメント

数値型配列のデータと文字型配列のストリングディクスリプタを格納します。実際の文字列はシンボルテーブルセグメントの文字列格納エリアに格納されます。

CLEAR文で大きさを変更することができます。

1つの配列変数について、1つのセグメントベースが割り当てられます。1つの配列変数で使用する大きさが64KBを超えることはありません。しかし、いくつかの配列変数を宣言することにより、64KBを超える領域を使用することができます。

(5) 機械語プログラムセグメント

BASICインタープリタが使用しない独立した領域です。この領域に機械語プログラムなどのデータを格納します。

CLEAR文で大きさを変更することができます。

D 文字コード表

下位ビット＼上位ビット	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0			スペース	0	@	P	`	p	▁	┴		ー	タ	ミ	═	╳
1			!	1	A	Q	a	q	▂	┬	。	ア	チ	ム	╞	円
2			"	2	B	R	b	r	▃	┤	「	イ	ツ	メ	╪	年
3			#	3	C	S	c	s	▄	├	」	ウ	テ	モ	╡	月
4			$	4	D	T	d	t	▅	▔	、	エ	ト	ヤ	◢	日
5			%	5	E	U	e	u	▆	─	・	オ	ナ	ユ	◣	時
6			&	6	F	V	f	v	▇	│	ヲ	カ	ニ	ヨ	◥	分
7	BEEP		'	7	G	W	g	w	█	▕	ァ	キ	ヌ	ラ	◤	秒
8	BS		(	8	H	X	h	x	▏	┌	ィ	ク	ネ	リ	♠	
9	TAB		)	9	I	Y	i	y	▎	┐	ゥ	ケ	ノ	ル	♥	
A	LF		*	:	J	Z	j	z	▍	└	ェ	コ	ハ	レ	♦	
B	HOME	ESC	+	;	K	[	k	{	▌	┘	ォ	サ	ヒ	ロ	♣	
C	CLR	→	,	<	L	¥	l	¦	▋	╭	ャ	シ	フ	ワ	●	╲
D	↵	←	−	=	M	]	m	}	▊	╮	ュ	ス	ヘ	ン	○	
E		↑	.	>	N	^	n	~	▉	╰	ョ	セ	ホ	゛	╱	
F		↓	/	?	O	_	o		┼	╯	ッ	ソ	マ	゜	╲	

E キー機能一覧表

キャラクタコード(10進)	対応するキー操作	機　能
1	CTRL + A	HELP キーと同じ
2	CTRL + B	カーソルを1つ前の単語の先頭へ移動
3	CTRL + C	STOP キーと同じ
4	CTRL + D	カーソルのある単語を削除
5	CTRL + E	カーソルのある行を削除
6	CTRL + F	カーソルを次の単語の先頭へ移動
7	CTRL + G	スピーカを鳴らす
8	CTRL + H	バックスペース(BS)
9	CTRL + I	水平タブ(HT)
10	CTRL + J	ラインフィード(LF)
11	CTRL + K	カーソルをホームポジションへ移動
12	CTRL + L	テキスト画面クリア
13	CTRL + M	キャリッジリターン(CR)
14	CTRL + N	カーソルを次の単語の先頭へ移動
15	CTRL + O	画面表示の無効/有効を切り換える
17	CTRL + Q	カーソルを行の先頭へ移動
18	CTRL + R	挿入モードに入る、挿入モードを抜ける
19	CTRL + S	表示を一時停止する
21	CTRL + U	カーソルのある行をキャンセル
24	CTRL + X	カーソルを行の最後へ移動
26	CTRL + Z	カーソルから画面終りまで消去
28	→	カーソルを右へ移動
29	←	カーソルを左へ移動
30	↑	カーソルを上へ移動
31	↓	カーソルを下へ移動

F 数値演算プロセッサ

EPSON PC シリーズに数値演算プロセッサを装着することにより次の数学関数の実行速度を速めることができます。(単精度実数、倍精度実数演算とも可)

^(べき乗)、ATN、COS、EXP、LOG、SIN、SQR、TAN

なお、四則演算は高速化しません。

数値演算プロセッサを取り付けた場合は、メモリスイッチの値も変更してください。

1．数値演算プロセッサの取り付け方法

各 EPSON PC シリーズのユーザーズマニュアルの「数値演算プロセッサ」の章を参照してください。

2．メモリスイッチの変更

メモリスイッチ	意味		
SW3-3	数値演算プロセッサの使用 (アドバンスドモード)	0	数値演算プロセッサを使用しない
		1	数値演算プロセッサを使用する
SW3-4	数値演算プロセッサの使用 (ノーマルモード)	0	数値演算プロセッサを使用しない
		1	数値演算プロセッサを使用する
SW3-5	数値演算プロセッサのクロック数	0	10MHz(PC286SEP または PCSEP2)
		1	8 MHz

- PC-286L/LE シリーズでは数値演算プロセッサを使用することはできません。
- PC-286U シリーズでは、ノーマルモードのみです。

G キースキャンコード

INP関数により、キーボードの押下状態を調べることができます。キースキャンのポートアドレスは16進数で &HEO から &HEC です。それぞれのポートで調べることのできるキーは次の通りです。

注意：このスキャンコードはBASICシステムが作成しているもので、他のシステムでこのポートからキーを押下状態を調べることはできません。

ポートアドレス＼ビット位置	7	6	5	4	3	2	1	0
E 0	7	6	5	4	3	2	1	0
E 1	. |	.	,	=	+	*	9	8
E 2	G キ	F ハ	E イ イ	D シ	C ソ	B コ	A チ	~ @ ゛
E 3	O ラ	N ミ	M モ	L ソ	K ノ	J マ	I ニ	H ク
E 4	W テ	V ヒ	U ナ	T カ	S ト	R ス	Q タ	P セ
E 5	= − ホ	6 . へ	] | 」 a	¥ | _	[| 「 °	Z ッ ツ	Y ン	X サ
E 6	' 7 ャ ヤ	& 6 ォ オ	% 5 ェ エ	$ 4 ゥ ウ	# 3 ァ ア	" 2 フ	! 1 ヌ	0 ヲ ワ
E 7	− ロ	? / ・ メ	> . 。 ル	< , 、 ネ	+ ; レ	* : ケ	) 9 ョ ヨ	(8 ュ ユ
E 8	CTRL	SHIFT	カナ	GRPH	DEL	→	↑	HOME CLR
E 9	ESC	(SPACE)	f･5	f･4	f･3	f･2	f･1	STOP
E A	CAPS	/	−	COPY	HELP	←	↓	TAB
E B	╳	╳	╳	╳	╳	╳	ROLL DOWN	ROLL UP
E C	INS	f･10	f･9	f･8	f･7	f･6	XFER	BS

例えばPRINT HEX$(INP(&HE0))を実行して表示する値から、押しているキーを調べることができます。

例

```
CSL 3 : WHILE -1 : LOCATE 0, 0 : PRINT HEX$(INP(&HE0)) : WEND ↵
```

&HFE(2進数で11111111)	どのキーも押していない
&HFE(11111110)	[0] を押している。
&HFD(11111101)	[1] を押している。
&HFB(11111011)	[2] を押している。
&HF7(11110111)	[3] を押している。
&HEF(11101111)	[4] を押している。
&HDF(11011111)	[5] を押している。
&HBF(10111111)	[6] を押している。
&H7F(01111111)	[7] を押している。

2個のキーを同時に押している場合は、2つのビットが0になります。例えば [1] と [2] のキーを同時に押せば結果は&HF9(11111001)になります。

また、すべてのキーの状態は、次のようにして調べることができます。

```
CSL 3 : WHILE -1 : LOCATE 0, 0 : FOR I=&HE0 TO &HEC : PRINT HEX$(INP(I)) :
NEXT : WEND ↵
```

索　　引

【数字】

【英字】

【あ】

【か】

【さ】

【ま】

【や】

ページ中の英字(HEX$, OCT$ など)は日本語 Disk BASIC リファレンスマニュアル中の該当ステートメントを示します。これらの用語についてはリファレンスマニュアルを参照してください。

【記号】

人と情報の接点をみつめる **EPSON**

●エプソンPCシリーズに関する技術的なご質問・ご相談に電話でお答えします。
エプソンPCインフォメーションセンター 東京(03)377-3531 大阪(06)397-0915
●受付時間/AM9:00〜PM5:30 月曜日〜金曜日(祝日を除く)

エプソン販売株式会社

●本　社:〒151 東京都渋谷区初台1-53-6 ●ショールーム:新宿NSビル5階

■支店・営業所

●札幌	(011)222-2821	●金沢	(0762)62-3216
●仙台	(022)263-3691	●静岡	(0542)51-1061
●秋田	(0188)32-4002	●名古屋	(052)962-7001
●酒田	(0234)23-8200	●京都	(075)361-7551
●大宮	(048)644-3400	●大阪	(06) 397-0900
●千葉	(0472)25-0984	●大阪南	(06) 632-3353
●東京	(03) 348-6801	●広島	(082)262-5181
●東京中央	(03) 258-4841	●高松	(0878)23-3646
●横浜	(045)316-4820	●福岡	(092)471-0761
●長野	(0262)24-7660	●鹿児島	(0992)25-7717
●松本	(0263)36-7251	●特販部	(03) 377-3321
●新潟	(025)243-8515		

※電話のかけまちがいが増えておりますので、番号をよくお確めの上おかけください。

■製品の修理に関するお問合わせは、下記サービスセンターまでお願いします。

●札幌サービスセンター 〒060 札幌市中央区北一条西2丁目札幌時計台ビル6階 (011)222-2821
●仙台サービスセンター 〒980 仙台市青葉区一番町4-1-1仙台セントラルビル4階 (022)263-3691
●東京サービスセンター 〒151 東京都渋谷区初台1-53-6 (03) 377-7001
●松本サービスセンター 〒390 松本市中央2-1-27松本本町第一生命ビル8階 (0263)36-7251
●名古屋サービスセンター 〒460 名古屋市中区新栄町2-13栄第一生命ビル9階 (052)962-7001
●大阪サービスセンター 〒532 大阪市淀川区宮原3-5-24新大阪第一生命ビル6階 (06) 397-0930
●広島サービスセンター 〒732 広島市東区光町1-12-16栄泉広島ビル5階 (082)262-5181
●福岡サービスセンター 〒812 福岡市博多区博多駅東2-6-23住友博多駅前第二ビル7階 (092)471-0761

●受付時間/AM9:00〜PM5:00 月曜日〜金曜日(祝日を除く)

セイコーエプソン株式会社

本　社 〒392 長野県諏訪市大和3-3-5

89.5.30

EPSON PC シリーズ日本語 Disk BASIC V3.0

ユーザーズマニュアル
1989年7月　第2版 第1刷発行
セイコーエプソン株式会社
広丘事業所　電子機器事業本部
〒399-07 長野県塩尻市広丘原新田80番地

お問い合わせは、エプソン販売㈱までお願いいたします。

EPSON
日本語 Disk BASIC

ご使用の前に

Y16199103800

このたびは日本語Disk BASICをお求めいただきまして、誠にありがとうございます。

日本語Disk BASICを梱包箱から取り出しましたら、下記のチェックリストをご覧になりながら、梱包内容をお確かめください。万一、不足している添付品がございましたら、お求めいただきました販売店までご連絡ください。

○チェックリスト

梱 包 内 容	チェック欄
日本語 Disk BASICシステムディスク	
日本語 Disk BASIC ユーティリティディスク	
日本語 Disk BASIC ユーザーズマニュアル	
日本語 Disk BASIC リファレンスマニュアル	
日本語 Disk BASIC ユーザー登録カード	

89. 07-50. 0